CLiC-it 2021
Italian Conference on Computational Linguistics

Proceedings of the Eighth Italian Conference on Computational Linguistics

Milan, Italy, January 26-28, 2022

Editors:

Elisabetta Fersini
Marco Passarotti
Viviana Patti

aA ccademia university press

aA

Accademia University Press

Proceedings of the Eighth Italian Conference on Computational Linguistics CliC-it 2021

Milan, Italy, 26-28 January, 2022

Elisabetta Fersini, Marco Passarotti and Viviana Patti (dir.)

DOI: 10.4000/books.aaccademia.10417
Publisher: Accademia University Press
Place of publication: Torino
Year of publication: 2022
Published on OpenEdition Books: 20 October 2022
Series: Collana dell'Associazione Italiana di Linguistica Computazionale
Electronic EAN: 9791280136947

OpenEdition Books
https://books.openedition.org

Printed version
Number of pages: 434

Electronic reference
FERSINI, Elisabetta (ed.) ; PASSAROTTI, Marco (ed.) ; and PATTI, Viviana (ed.). *Proceedings of the Eighth Italian Conference on Computational Linguistics CliC-it 2021: Milan, Italy, 26-28 January, 2022.* New edition [online]. Torino: Accademia University Press, 2022 (generated 24 octobre 2022). Available on the Internet: <http://books.openedition.org/aaccademia/10417>. ISBN: 9791280136947. DOI: https://doi.org/10.4000/books.aaccademia.10417.

ABSTRACTS

The eighth edition of the Italian Conference on Computational Linguistics (CLiC-it 2021) was held at Università degli Studi di Milano-Bicocca from 26th to 28th January 2022.
After the edition of 2020, which was held in fully virtual mode due to the health emergency related to Covid-19, CLiC-it 2021 represented the first moment for the Italian research community of Computational Linguistics to meet in person after more than one year of full/partial lockdown.

ELISABETTA FERSINI (DIR.)

Università degli Studi di Milano-Bicocca Viale Sarca 336, Ed. U14, 20126 Milano - fersiniel[@]disco.unimib.it

MARCO PASSAROTTI (DIR.)

CIRCSE Research Centre Università Cattolica del Sacro Cuore Largo Gemelli, 1 – 20123 Milan, Italy marco.passarotti[@]unicatt.it

VIVIANA PATTI (DIR.)

University of Torino Italy - patti[@]di.unito.it

sede legale: c/o Bernardo Magnini, Via delle Cave 61, 38122 Trento
codice fiscale 96101430229
email: info@ai-lc.it

Accademia University Press
via Carlo Alberto 55
I-10123 Torino
info@aAccademia.it

isbn 9791280136770
www.aAccademia.it/CLIC_2021

Accademia University Press è un marchio registrato di proprietà
di LEXIS Compagnia Editoriale in Torino srl

Table of Contents

Contributed Papers: Short Papers

Preface

The eighth edition of the Italian Conference on Computational Linguistics (CLiC-it 2021) was held at Università degli Studi di Milano-Bicocca from 26th to 28th January 2022.

After the edition of 2020, which was held in fully virtual mode due to the health emergency related to Covid-19, CLiC-it 2021 represented the first moment for the Italian research community of Computational Linguistics to meet in person after more than one year of full/partial lockdown. Although the conference was held in dual mode, we strongly suggested the participants to attend it coming to Milan. Indeed, we received a strong feedback on this aspect from the community, which was eager to meet in person and enjoy both the scientific and social events together with the colleagues. In total, 99 participants registered to the conference benefiting from the early registration fee, 91 out of which expressed their intention to attend the event in person, which we consider as a very positive indication of enthusiasm from the community, given the uncertain situation due to the evolution of the pandemic in Italy.

In total, we received 68 proposals, organized in the following specific tracks: Information Extraction, Information Retrieval and Question Answering, Computational Linguistics and Natural Language Processing for the Humanities, Computational Social Science and Social Media Dialogue, Discourse and Natural Language Generation, Ethics and NLP, Language Resources and Evaluation, Spoken Language Processing and Automatic Speech Understanding, Cognitive Modeling and Psycholinguistics, Linguistic Issues in CL and NLP, Machine Learning for NLP, Machine Translation and Multilingualism, Morphology and Syntax Processing, Pragmatics and Creativity, Research and Industrial NLP Applications, Semantics, Knowledge Representation, Vision, Robotics, Multimodal and Grounding.

During the reviewing process, each submission was reviewed by three independent members of the scientific committees of the tracks in single-blind fashion. At the end of the process, 59 proposals were accepted for presentation at the conference and publication in the proceedings, resulting in an acceptance rate of 86.76%. Out of the 59 accepted proposals, 26 were included in the program of CLiC-it 2021 as oral presentations and the remaining 33 were assigned to one of the three poster sessions of the conference. As usual, the criterion for assigning a proposal to an oral or a poster session was based on the contents and not on the quality of the proposal. Regardless of the format of presentation, all accepted papers are allocated six pages of content (one additional page w.r.t. the previous editions) plus unlimited pages of references in the proceedings, available as open access publication. In line with last editions, the conference is receiving considerable attention from the international community, with 21 (31%) submissions this year showing at least one author affiliated to a foreign institution. This amounts to a total of 43 authors over 221 (20%) affiliated to 13 foreign countries: Belgium, Czechia, France, Germany, Romania, Hong Kong, Greece, Iran,The Netherlands, Spain, Switzerland, Turkey, United States.

The program of CLiC-it 2021 is completed by 10 research communications selected after the reviewing process. Research communications are not published in the proceedings, but have been orally presented within dedicated sessions at the conference, in order to enforce dissemination of excellence in research.

In addition to the technical programme, this year we were honoured to have as invited speakers such internationally recognized researchers as Barbara Plank (IT University of Copenhagen), with a keynote entitled "Returning the L in NLP: Why Language (Variety) Matters and How to Embrace it in Our

Models" and Dan Roth (University of Pennsylvania & Amazon AWS AI) with the keynote "It's Time for Reasoning".

Two tutorials were organized. On 26th January, Julia Bosque Gil (University of Zaragoza, Spain) provided an introduction to Linguistic Linked Data, while on 28th January Daniel Zeman (Charles University of Prague, Czech Republic) introduced the project of Universal Dependencies.

Moreover, the program included a panel discussion about linguistic infrastructures with representatives of CLARIN (Monica Monachini, ILC-CNR), ELG (Bernardo Magnini, FBK), and DARIAH (Emiliano Degl'Innocenti, OVI-CNR).

This year we have received 8 candidate theses for the "Emanuele Pianta Award for the Best Master Thesis". This special prize for the best Master Thesis (Laurea Magistrale) in Computational Linguistics, submitted at an Italian University, is endorsed by AILC. The candidate theses have been evaluated by a special jury, and awarded during the last session of the conference, by the members of the jury.

We thank all the people and institutions involved in the organization of the conference, all track chairs, reviewers, and all participants, who contributed to the success of the event. All track chairs and reviewers are named in the following pages. We are grateful to the following, who made CLiC-it 2021 possible and supported us greatly and generously in the processes of local organization, publication of the proceedings and publicity: Marco Cremaschi, Giulia Rizzi, Silvia Terragni (Local Organizing Committee); Alessandra Teresa Cignarella, Matteo Pellegrini, Danilo Croce (Publication Chairs); Rachele Sprugnoli (Publicity Chair).

We thank the sponsors of CLiC-it 2021, who generously provided funds and services that are crucial for the realization of this event, including: B13, Dipartimento di Informatica, Sistemistica e Comunicazione (DISCO), Università degli Studi di Milano-Bicocca, Elsevier, Red Software Systems, Yewno (Gold sponsors); AIO Proactive Systems, CELI, expert.ai (Silver sponsors); datrix, ELRA, oaxs (Bronze sponsors).

Finally, we want to thank very much the Associazione Italiana di Linguistica Computazionale (AILC), all the members of the Association Board and, in particular, the President Bernardo Magnini, who never let us alone in the sea of doubts and problems that organizing an hybrid event implies.

December 2021

Elisabetta Fersini, Marco Passarotti, Viviana Patti
CLiC-it 2021 General Chairs

Conference and Programme Chairs

Elisabetta Fersini, Università degli Studi di Milano-Bicocca, Italy
Marco Passarotti, Università Cattolica del Sacro Cuore, Milano, Italy
Viviana Patti, Università degli Studi di Torino, Italy

Track Chairs

Information Extraction, Information Retrieval and Question Answering
Pierpaolo Basile, Università degli Studi di Bari
Elena Cabrio, Université Côte d'Azur

CL and NLP for the Humanities
Francesca Frontini, Istituto di Linguistica Computazionale "A.Zampolli"
Francesco Mambrini, Università Cattolica del Sacro Cuore, Milano

Computational Social Science and Social Media
Valerio Basile, Università degli Studi di Torino
Nicole Novielli, Università degli Studi di Bari

Dialogue, Discourse and Natural Language Generation
Alessandro Mazzei, Università degli Studi di Torino
Antonio Origlia, Università degli Studi di Napoli Federico II

Ethics and NLP
Andrea Bolioli, CELI Language Technology
Malvina Nissim, University of Groningen

Language Resources and Evaluation
Simonetta Montemagni, Istituto di Linguistica Computazionale "A.Zampolli"
Rachele Sprugnoli, Università Cattolica del Sacro Cuore, Milano

Spoken Language Processing and Automatic Speech Understanding
Franco Cutugno, Università degli Studi di Napoli Federico II
Daniele Falavigna, Fondazione Bruno Kessler

Cognitive Modeling and Psycholinguistics
Cristiano Chesi, Scuola Superiore Universitaria Pavia – IUSS
Tommaso Fornaciari, Università Bocconi

Linguistic Issues in CL and NLP
Maria Teresa Guasti, Università degli Studi di Milano-Bicocca
Elisabetta Jezek, Università di Pavia

Machine Learning for NLP
Roberto Basili, Università di Roma, Tor Vergata
Felice Dell'Orletta, Istituto di Linguistica Computazionale "A.Zampolli"

Machine Translation and Multilingualism
Matteo Negri, Fondazione Bruno Kessler
Manuela Sanguinetti, Università degli Studi di Cagliari

Morphology and Syntax Processing
Cristina Bosco, Università degli Studi di Torino
Fabio Tamburini, Università di Bologna

Pragmatics and Creativity
Valentina Bambini, Scuola Superiore Universitaria Pavia – IUSS
Carlo Strapparava, Fondazione Bruno Kessler

Research and Industrial NLP Applications
Francesca Bonin, IBM Research Ireland
Marcello Federico, Amazon

Semantics, Knowledge Representation
Alessandro Lenci, Università di Pisa
Fahad Khan, Istituto di Linguistica Computazionale "A.Zampolli"

Vision, Robotics, Multimodal and Grounding
Raffaella Bernardi, Università di Trento
Danilo Croce, Università di Roma, Tor Vergata

Local Organizing Committee

Marco Cremaschi, Università degli Studi di Milano-Bicocca
Giulia Rizzi, Università degli Studi di Milano-Bicocca
Silvia Terragni, Università degli Studi di Milano-Bicocca

Publication Chairs

Alessandra Teresa Cignarella, Università degli Studi di Torino
Matteo Pellegrini, Università Cattolica del Sacro Cuore, Milano
Danilo Croce, Università di Roma, Tor Vergata

Publicity Chair

Rachele Sprugnoli, Università Cattolica del Sacro Cuore, Milano

Reviewers

Luca Anselma, Alessio Palmero Aprosio, Mattia Atzeni, Stefano Bannò, Pierpaolo Basile, Valerio Basile, Luisa Bentivogli Monica Berti, Arianna Bisazza, Luca Bischetti, Yuri Bizzoni Paolo Boldi, Andrea Bolioli, Marianna Bolognesi, Cristina Bosco, Davide Buscaldi, Elena Cabrio, Tommaso Caselli, Pierluigi Cassotti, Giuseppe Castellucci, Flavio Massimiliano Cecchini, Cristiano Chesi, Francesca Chiusaroli, Alessandra Teresa Cignarella, Andrea Cimino, Giovanni Colavizza, Davide Colla, Danilo Croce, Francesco Cutugno, Rossana Damiano, Lorenzo De Mattei, Thierry Declerck, Felice Dell'Orletta, Rodolfo Delmonte, Danilo Dessì, Luigi Di Caro Mattia Antonino Di Gangi, Maria Di Maro, Giorgio Maria Di Nunzio, Claudio Fantinuoli, Simone Filice, Komal Florio, Greta Franzini, Diego Frassinelli, Simona Frenda, Francesca Frontini, Gloria Gagliardi, Marco Gaido, Emiliano Giovannetti, Martin Gleize, Claudio Greco, Shohreh Haddadan, Christian Hardmeier, Sadid A. Hasan, Aurelie Herbelot Delia Irazu Hernandez Farias, Ignacio Iacobacci Andrea Iovine, Elisabetta Jezek, Charles Jochim, Roman Klinger, Alberto Lavelli, Matteo Magnani, Mirko Marras, Barbara McGillivray, Stefano Menini, Alessio Miaschi, Monica Monachini, Massimo Moneglia, Johanna Monti, Alessandro Moschitti, Mohamed Nabih, Debora Nozza, Renáta Németh, Antonio Origlia, Petya Osenova, Alessio Palmero Aprosio, Alessandro Panunzi, Sara Papi, Patrick Paroubek, Matteo Pellegrini, Michael Piotrowski, Marco Polignano, Valeria Quochi, Alessandro Raganato, Andrea Amelio Ravelli, Matteo Romanello, Marco Rospocher, Gaetano Rossiello, Irene Russo, Manuela Sanguinetti, Gabriele Sarti, Beatrice Savoldi, Flavia Sciolette, Lucia Siciliani, Gianmaria Silvello, Maria Simi, Manuela Speranza, Francesca Spezzano, Rachele Sprugnoli, Armando Stellato, Fabio Tamburini, Mariona Taulé, Sara Tonelli, Rocco Tripodi, Marieke van Erp, Rossella Varvara, Giulia Venturi, Serena Villata.

CLiC-it 2021 is sponsored by

Gold Sponsors

Silver Sponsors

Bronze Sponsors

Keynote Talks and Tutorial

Returning the L in NLP: Why Language (Variety) Matters and How to Embrace it in Our Models

Barbara Plank
Computer Science Department
IT University of Copenhagen

Abstract

NLP's success today is driven by advances in modeling together with huge amounts of unlabeled data to train language models. However, for many application scenarios like low-resource languages, non-standard data and dialects we do not have access to labeled resources and even unlabeled data might be scarce. Moreover, evaluation today largely focuses on standard splits, yet language varies along many dimensions [3]. What is more is that for almost every NLP task, the existence of a single perceived gold answer is at best an idealization.

In this talk, I will emphasize the importance of language variation in inputs and outputs and its impact on NLP. I will outline ways on how to go about it. This includes recent work on how to transfer models to low-resource languages and language variants [5, 6], the use of incidental (or fortuitous) learning signals such as genre for dependency parsing [2] and learning beyond a single ground truth [1, 3, 4].

Biography. Barbara Plank is Professor in the Computer Science Department at ITU (IT University of Copenhagen). She is also the Head of the Master in Data Science Program. She received her PhD in Computational Linguistics from the University of Groningen. Her research interests focus on Natural Language Processing, in particular transfer learning and adaptations, learning from beyond the text, and in general learning under limited supervision and fortuitous data sources. She (co)-organised several workshops and international conferences, amongst which the PEOPLES workshop (since 2016) and the first European NLP Summit (EurNLP 2019). Barbara was general chair of the 22nd Northern Computational Linguistics conference (NoDaLiDa 2019) and workshop chair for ACL in 2019. Barbara is member of the advisory board of the European Association for Computational Linguistics (EACL) and vice-president of the Northern European Association for Language Technology (NEALT).

References

[1] Tommaso Fornaciari, Alexandra Uma, Silviu Paun, Barbara Plank, Dirk Hovy, and Massimo Poesio. Beyond Black & White: Leveraging Annotator Disagreement via Soft-Label Multi-Task Learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2591–2597, 2021.

[2] Max Müller-Eberstein, Rob van der Goot, and Barbara Plank. Genre as Weak Supervision for Cross-lingual Dependency Parsing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4786–4802, 2021.

[3] Barbara Plank. What to do about non-standard (or non-canonical) language in NLP. In *Proceedings of KONVENS 2016, Ruhr-University Bochum*. Bochumer Linguistische Arbeitsberichte, 2016.

[4] Barbara Plank, Dirk Hovy, and Anders Søgaard. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 742–751, 2014.

[5] Barbara Plank, Kristian Nørgaard Jensen, and Rob van der Goot. DaN+: Danish nested named entities and lexical normalization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6649–6662, 2020.

[6] Rob van der Goot, Ibrahim Sharaf, Aizhan Imankulova, Ahmet Üstün, Marija Stepanović, Alan Ramponi, Siti Oryza Khairunnisa, Mamoru Komachi, and Barbara Plank. From Masked Language Modeling to Translation: Non-English Auxiliary Tasks Improve Zero-shot Spoken Language Understanding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2479–2497, 2021.

It's Time for Reasoning

Dan Roth
University of Pennsylvania & Amazon AWS AI

Abstract

The fundamental issue underlying natural language understanding is that of semantics – there is a need to move toward understanding natural language at an appropriate level of abstraction in order to support natural language understanding and communication with computers. Machine Learning has become ubiquitous in our attempt to induce semantic representations of natural language and support decisions that depend on it; however, while we have made significant progress over the last few years, it has focused on classification tasks for which we have large amounts of annotated data. Supporting high level decisions that depend on natural language understanding is still beyond our capabilities, partly since most of these tasks are very sparse and knowledge-intensive, and generating supervision signals for it does not scale. I will discuss some of the challenges underlying reasoning – making natural language understanding decisions that depend on multiple, interdependent, models, and exemplify it mostly using the domain of Reasoning about Time, as it is expressed in natural language.

Biography. Dan Roth is the Eduardo D. Glandt Distinguished Professor at the Department of Computer and Information Science, University of Pennsylvania, lead of NLP Science at Amazon AWS AI, and a Fellow of the AAAS, the ACM, AAAI, and the ACL. In 2017 Roth was awarded the John McCarthy Award, the highest award the AI community gives to mid- career AI researchers. Roth was recognized "for major conceptual and theoretical advances in the modeling of natural language understanding, machine learning, and reasoning." Roth has published broadly in machine learning, natural language processing, knowledge representation and reasoning, and learning theory, and has developed advanced machine learning based tools for natural language applications that are being used widely. Roth was the Editor-in-Chief of the Journal of Artificial Intelligence Research (JAIR) and a program chair of AAAI, ACL, and CoNLL. Roth has been involved in several startups; most recently he was a co-founder and chief scientist of NexLP, a startup that leverages the latest advances in Natural Language Processing (NLP), Cognitive Analytics, and Machine Learning in the legal and compliance domains. NexLP was acquired by Reveal in 2020. Prof. Roth received his B.A Summa cum laude in Mathematics from the Technion, Israel, and his Ph.D. in Computer Science from Harvard University in 1995.

Introduction to Linguistic Linked Data

Julia Bosque-Gil
Aragon Institute of Engineering Research
University of Zaragoza

Abstract

The number of resources that provide lexical data keeps increasing as outcomes of projects in (computational) linguistics, digital humanities, and e-lexicography. This vast landscape of heterogeneous and often isolated language resources creates obstacles for their straightforward linking and integration in pipelines in an interoperable manner. To address this, experts working in the domain of the Semantic Web have adopted approaches to linguistic data representation based on the Linked Data (LD) paradigm, giving birth to the Linguistic Linked Data (LLD) line of research. In this context, linked data emerges as a way to make linguistic data uniformly query-able, interoperable, and easily discoverable as well as reusable on the basis of web standards. This tutorial will provide attendees a theoretical and practical overview of the foundations of LLD, covering, among other aspects, an introduction to the Semantic Web and linked data, and a walkthrough of the different steps for linguistic linked data generation. We will lay special emphasis on knowledge representation with the de-facto standard for lexical data representation on the Web, the OntoLex-Lemon model, and other linguistic vocabularies. Participants are encouraged to select language resources of their interest in advance to address their modelling as part of the practical session.

Biography. Dr. Julia Bosque-Gil is a postdoctoral researcher at the Distributed Information Systems Group at University of Zaragoza and a member of the Aragon Institute of Engineering Research (I3A). For her PhD she investigated the use of linguistic linked data for lexicography, and currently works on the representation and linking of multilingual resources as linguistic linked data as part of the Prêt-à-LLOD project. She co-leads NexusLinguarum COST Action Working Group 1 on Linked-data based Language Resources and co-chairs the Ontology-Lexica Community Group since October 2018.

Universal Dependencies: Principles and Tools

Daniel Zeman
Institute of Formal and Applied Linguistics
Charles University, Czech Republic
Faculty of Mathematics and Physics

Abstract

Universal Dependencies is an international community project and a collection of morphosyntactically annotated data sets ("treebanks") for more than 100 languages. The collection is an invaluable resource for various linguistic studies, ranging from grammatical constructions within one language to language typology, documentation of endangered languages, and historical evolution of language. In the tutorial, I will first quickly show the main principles of UD, then I will present the actual data and various tools that are available to work with it: parsers, batch processors, search engines and viewers.

Biography. Daniel Zeman is a senior researcher and lecturer at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University in Prague. His research interests range from natural language parsing and morphology to low-resource language processing and linguistic typology. He is one of the co-founders and main coordinators of the Universal Dependencies project.

CEUR-WS.org/Vol-3033/tutorial2.pdf

Contributed Papers: Long Papers

Atypical or Underrepresented? A Pilot Study on Small Treebanks

Akshay Aggarwal[1] **and Chiara Alzetta**[2]
1. Twilio, Prague, Czechia
2. Istituto di Linguistica Computazionale "A.Zampolli", CNR, Pisa - ItaliaNLP Lab
aaggarwal@twilio.com, chiara.alzetta@ilc.cnr.it

Abstract

We illustrate an approach for multilingual treebanks explorations by introducing a novel adaptation to small treebanks of a methodology for identifying cross-lingual quantitative trends in the distribution of dependency relations. By relying on the principles of cross-validation, we reduce the amount of data required to execute the method, paving the way to expanding its use to low-resources languages. We validated the approach on 8 small treebanks, each containing less than 100,000 tokens and representing typologically different languages. We also show preliminary but promising evidence on the use of the proposed methodology for treebank expansion.

1 Introduction and Motivation

Linguistically-annotated language resources like treebanks are fundamental for developing reliable models to train and test tools used to address Natural Language Processing (NLP) tasks acquiring linguistic evidence from corpora. Concerning the latter, researchers frequently rely on multilingual or parallel resources in contrastive studies to quantify the similarities and differences between languages (Jiang and Liu, 2018). Over the past few years, the Universal Dependencies (UD) initiative[1] (Zeman et al., 2021) has further encouraged such studies. UD defines a universal inventory of categories and guidelines to facilitate consistent annotation of similar constructions across languages (Nivre, 2015; de Marneffe et al., 2021), and, at present, the project includes about 200 treebanks representing over 100 languages. The consistent annotation of linguistic phenomena under a shared representation and across different languages makes UD treebanks exceptionally well suited for quantitative comparison of languages (see, for example, Croft et al. (2017), Berdicevskis et al. (2018), Vylomova et al. (2020) and among our works, Alzetta et al. (2019a) and Alzetta et al. (2020a)).

Despite their great relevance for linguistic investigations, large treebanks are available for only a tiny fraction of the world's languages (Vania et al., 2019). Even within the UD project, around 60% of the treebanks can be considered small, i.e. containing less than 100,000 tokens. Treebank size, in fact, is generally identified as the bottleneck for obtaining high-quality representative models of language use to be employed in downstream NLP applications. In general terms, larger datasets allow for better generalisations of language constructions, leading to better performances of systems trained using such data (Zeman et al., 2018). In fact, *ad-hoc* strategies are generally needed when dealing with low-resourced languages (Hedderich et al., 2021).

This paper illustrates a novel workflow specifically designed to adapt an existing methodology for treebank exploration to small treebanks. The base method, extensively described by Alzetta et al. (2020b), relies on an unsupervised algorithm called *LISCA* (*LInguistically–driven Selection of Correct Arcs*) (Dell'Orletta et al., 2013). LISCA has been successfully employed in past works for performing quantitative cross-lingual analyses (Alzetta et al., 2019a; Alzetta et al., 2019b; Alzetta et al., 2020a) and error detection on UD treebanks (Alzetta et al., 2017). The algorithm works in two main steps. First, it acquires evidence about language use from the distributions of phenomena in annotated sentences. The algorithm then uses such evidence to distinguish *typical* from *atypical constructions* in an unseen set of sentences. The

[1] https://universaldependencies.org

typicality of a construction is determined with respect to the examples observed in a corpus used as a reference, and is encoded with a score. This score, in fact, reflects the probability of observing a dependency occurring in a given context (both sentence-level and corpus-level) on the basis of the constructions sharing common properties reported in the reference corpus. Hence, from our point of view, typicality and frequency are tightly related concepts, as non-standard constructions are also usually less frequent in natural language use.

As such, the LISCA methodology relies on large sets of automatically parsed sentences to collect the statistics about phenomena distributions: even if the data contains parsing errors[2], the corpus size guarantees the collected statistics reflect the actual language use. However, such an approach can be employed only for analysing languages for which large amounts of data are available, or at least for which the parser outputs are generally considered reliable. To overcome such a limit, Aggarwal (2020) suggested that if the statistics are acquired from gold annotations (such as treebanks), the algorithm could collect the statistics from fewer data since these resources are assumed to be error-free.

We implemented this proposal by adapting the original LISCA workflow as detailed in Section 2. Our variation to the original methodology is inspired by the k-fold approach commonly used for performing systems' cross-validation: according to this approach, a dataset is split into sub-sets of equal size, iteratively used for training and/or evaluating a system. We employ a similar strategy for evaluating the typicality of the dependency relations in each treebank split, acquiring the statistics from the sentences contained in the other splits rather than from an external reference corpus. This small but substantial change in the method workflow allows us to apply the LISCA algorithm to small treebanks, which is particularly relevant in the case of analyses performed on low-resource languages.

We tested the methodology in a case study, reported in Section 3, involving 8 languages represented using UD treebanks. Our goal is to test if our method can support linguistic investigations for exploring and quantifying similarities and differences between typologically different languages. To this aim, we first validate the adaptation to the original LISCA approach proposed here in Section 3.1. Then, we exemplify how the obtained results can be employed for linguistic investigations in Section 3.2. To improve the cross–linguistic comparability of the analysis, we relied on Parallel UD (PUD) treebanks: a collection of parallel treebanks developed for the CoNLL–2017 Shared Task on multilingual parsing (Zeman et al., 2017) and linguistically annotated under the UD representation. Being parallel, PUDs are particularly well suited for carrying out multilingual studies since they contain only 1,000 sentences manually translated from English into the other languages, representing a perfect testbed for our approach.

Before concluding the paper in Section 5, we report the results of preliminary investigations to explore whether our approach could also be employed for automatically identifying underrepresented phenomena in treebanks. Søgaard (2020) and Anderson et al. (2021) argue that some treebanks cover only a restricted sample of the structures commonly used in a language, leaving out less common phenomena. This *leakiness* might affect the performances of NLP systems even more than the system architecture. Thus, treebanks should be expanded not only to improve their representativeness but also to obtain more truthful performances of systems trained using them. Section 4 investigates if our methodology can contribute to this issue by exploring its application in automatic treebank expansion.

The **contributions** of the paper can be listed as: *(i)* a novel approach specifically designed for carrying out multilingual investigations on small treebanks; *(ii)* a case study involving eight typologically different languages to test the methodology; and *(iii)* a novel formula, introduced in Section 3.2, to measure the distance between dependents and their syntactic head which improves the cross-lingual comparability of treebanks with respect to such property.

2 Approach

The method presented in this paper relies on a methodology for treebank exploration based on the unsupervised algorithm LISCA (Dell'Orletta et al., 2013), which we adapted to expand its usage for small treebanks, namely containing less than

[2] An assumption when producing automatically parsed data is that most of the errors made by a parser are consistent. As we showed in (Alzetta et al., 2017), the LISCA-based method allows to spot these errors types in annotations.

100,000 tokens.

As mentioned earlier, LISCA can be employed to quantify the typicality of each dependency relation (hereafter *deprel*)[3] of a linguistically annotated corpus with respect to a large set of examples taken as reference (Alzetta et al., 2020b). To achieve this goal, the algorithm first collects statistics about linguistically motivated properties of *deprels* extracted from a corpus of automatically parsed sentences (called *reference corpus*) to create a statistical model (SM). Then, the algorithm calculates a typicality score for each *deprel* appearing in a *test corpus* relying on the SM while also considering its linguistic context to assess the relevance of the *dependency label* used for marking the *dependency* in the given context. When interpreting the assigned LISCA score, a *deprel* marked by LISCA as highly typical was possibly frequently observed in similar contexts also in the reference corpus. In contrast, an atypical *deprel* could be characterised by certain properties which make it somehow distant from the other instances of *dependency* marked with the same *label* in the reference corpus.

In essence, LISCA computes the score for a given *deprel* taking into account local properties (e.g., dependency length and direction) of each *deprel* in the test corpus as well as the linguistic context where it is located (e.g., distance form root, leaves and number of siblings), comparing them both against the properties and contexts of all *dependencies* annotated with the same *dependency label* in the reference corpus. For this reason, the reference corpus has generally corresponded to a large corpus of around 40M tokens: the corpus size allows accounting for a more comprehensive set of examples of linguistic constructions while also compensating for possible parser errors.

Workflow. For this study, we implemented the adaptation of the LISCA workflow proposed by Aggarwal (2020). Inspired by the k-fold validation approach, we modified the original approach as follows:

1) Split a treebank into k portions of equal size ($k = 4$ for this work), each containing the same number of sentences;
2) Use LISCA to acquire the statistics (encoded in the SM) about the distribution of linguistic phenomena from a reference corpus obtained by merging $k-1$ portions of the previously split treebank;
3) Use the obtained SM to compute the typicality score of the *deprels* appearing in the remaining treebank portion (i.e., the one not included in the reference corpus);
4) Repeat steps 2 and 3 until all k portions are analysed;
5) Merge the analysed portions and order the *deprels* by decreasing LISCA score to have a unique ranking of all the *deprels* in the treebank.

The ordered ranking of *deprels* can be explored to investigate which linguistic constructions, represented by means of the *deprels*, were marked as typical or atypical, characterised by higher and lower scores, respectively.

2.1 Data and Languages

We tested our method on a selection of Parallel UD (PUD) treebanks (Zeman et al., 2017), each containing 1,000 sentences. In order to encompass different language families and genera[4], we carried out the case study on the following eight languages: **Arabic** (AR; Afro-Asiatic, Semitic), **Czech** (CZ; Indo-European, Slavic), **English** (EN; Indo-European, Germanic), **Hindi** (HI; Indo-European, Indic), **Finnish** (FI; Uralic, Finnic), **Indonesian** (ID; Austronesian, Malayo-Sumbawan), **Italian** (IT; Indo-European, Romance) and **Thai** (TH; Tai-Kadai, Kam-Tai).

3 Results

3.1 Validating the Approach

We report the results of an analysis to verify whether the adapted and original LISCA-based methods return comparable results. To this aim, we compared the LISCA ranking of PUD deprels obtained using the original algorithm workflow, which employs a large reference corpus to build the language SM, and the novel workflow defined above, which acquires the statistics from the treebank itself. We carried out this analysis for Italian and English PUD treebanks. We manually verified in previous studies that the original approach applied to those languages allows capturing elements of linguistic and parsing complexity

[3]Given a deprel $A \xrightarrow{nsubj} B$, we refer to $A \rightarrow B$ as the *dependency*, with $nsubj$ as the *dependency label*.

[4]The language family and genus, reported between parenthesis as (ISO language code, family, genus), are acquired from the World Atlas of Language Structures (WALS, available online `https://wals.info/languoid`) (Dryer and Haspelmath, 2013).

distinguishing between typical and atypical constructions along with the produced ranking of *deprels* (Alzetta et al., 2019a; Alzetta et al., 2020b).

We compared the *deprel* rankings obtained using the two methodology workflows in terms of Spearman correlation, which returns a rank correlation coefficient indicating a statistical dependence between the rankings of two observed variables. The analysis showed a strong and significant correlation between the rankings produced relying on the two workflows in both languages. Specifically, we obtained a Spearman correlation coefficient of 0.95 ($p < 0.5$) for Italian and English.

Such high correlations confirm that gold corpora, although small, can be used to acquire relevant statistics about language use. Manually revised data might be limited in size. However, their annotations are also generally correct in the case of rare phenomena, which a parser could wrongly annotate due to their low frequency in the data. While large reference corpora compensate for the possibly wrong parses assigned to rare constructions with their size, small reference corpora shall compensate with consistency and correctness. Hence, we could say that using gold data for building the SM allows reducing the number of examples for acquiring language statistics. We notice a difference between the two rankings only when focusing on the bottom part, where we find *deprels* with the lowest scores. While the original method produces only a tiny number of *deprels* with LISCA score equal to 0, which we usually excluded from the analyses, we observe many more of them in the ranking produced with our workflow adaptation. LISCA score zero is assigned to those dependencies never observed in the reference corpus; thus, their typicality is extremely low. It is not surprising that smaller reference corpora produce a higher number of these cases, given their limited coverage. However, the high correlation coefficient reported above suggests that such *deprels* are still interesting from a linguistic perspective. They correspond to rare constructions in the language, obtaining a score slightly higher than zero in the case of a larger reference corpus but are still placed in the lower positions of the ranking.

$$LL_{adjusted} = \begin{cases} \dfrac{LL_{raw} \cdot exp\left(1 - \frac{TrbAvgSentLen}{SentLength}\right)}{SentLength} & \text{if } \frac{SentLength}{TrbAvgSentLen} < 0.5 \\ \dfrac{LL_{raw}}{SentLength} & \text{if } \frac{SentLength}{TrbAvgSentLen} \in [0.5, 1.25] \text{ AND} \\ & LL_{raw} < \lfloor TrbAvgSentLen \rfloor \\ min\left(1, \frac{LL_{raw}}{TrbAvgSentLen}\right) & \text{otherwise} \end{cases}$$

Figure 1: $LinkLengthAdjusted$ formula for normalising *deprel length* in multilingual comparisons. **Note:** $\lfloor \cdot \rfloor$ denotes floor function, while $[a, b]$ denotes closed interval over a and b.

3.2 Rankings Exploration

This subsection exemplifies how the ranking of *deprels* obtained with our adapted approach can be employed in linguistic analyses to identify similarities and differences between languages. For this case study, we focused on a specific property of *deprels*, namely the *length* of the dependency link. The length of a *deprel*, measured as the linear distance in terms of intervening tokens between a word and its syntactic head, is a property frequently explored in linguistically annotated corpora. It is highly related to processing complexity in all languages (Demberg and Keller, 2008; Temperley, 2007; Futrell et al., 2015; Yu et al., 2019). For example, McDonald and Nivre (2011) observed that parsers tend to make more mistakes on longer sentences and longer dependencies. Such complexity makes this property particularly interesting from a multilingual perspective, especially when dealing with parallel corpora, as in our case study.

We inspected the ranking of *deprels* to monitor the LISCA score associated with *deprels* of different lengths and their distribution along the ranking of each language. To facilitate the rankings exploration and comparison, we split each ranking into three portions of equal size, referred to as *top*, *middle* and *bottom*, where *top* contains deprels obtaining the highest scores (more typical). In contrast, the *bottom* contains the *deprels* with the lowest scores (atypical).

In order to allow a proper multilingual comparison of the distribution of *deprel lengths* along with the rankings, we defined the novel measure called **Adjusted Link Length** ($LL_{adjusted}$, cf. Figure 1). The measure, inspired by Brevity Penalty used in BLEU score (Papineni et al., 2002), is designed to compute the length of *deprels* involving content words as dependant while simultaneously improving cross-language comparability as the length of

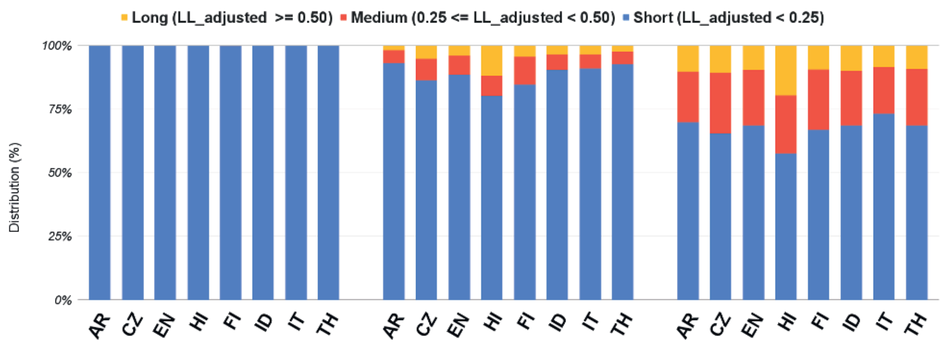


Figure 2: Distribution of Adjusted Link Length on content words across LISCA Rankings.

a *deprel* is measured keeping in mind the overall length of the sentence where it is located and the average sentence length in the treebank. This way, instead of comparing absolute length values, we can observe the tendency of languages towards producing longer or shorter *deprels*.

In $LL_{adjusted}$, we operationally compute the length of *deprels* as a function of *a)* the average sentence length in the treebank ($TrbAvgSentLen$), *b)* the length of the sentence where the *deprel* appears ($SentLength$), and *c)* the distance, in tokens, between the dependent and its syntactic head (LL_{raw}). The formula's values of 0.5 and 1.25 were determined empirically to account for unusually short and long sentences, respectively, in the treebank. Thus, the resulting value associated with each *deprel* denotes it as 'long', 'medium' or 'short' with respect to the average *deprel length* computed in the treebank. Note that, although our analysis focuses on content words, function words are still accounted for when computing the LISCA score as they might be part of the context of content words.

Figure 2 displays the distribution of *deprels* of different lengths (computed using $LL_{adjusted}$) along the portions of the treebank ranking of each language. The distributions show that longer *deprels* are given a lower plausibility score by LISCA in all languages. Interestingly, the length distributions are pretty similar across different languages except for Hindi. Such difference could be due to the typical word order of constituents of the considered languages. Hindi, in fact, is the only language of our set where the order of the main constituents is of the type S(ubject)O(bject)V(erb)[5], and the dominant word order of a language has been shown to influence the dependency length across major dependency types by Yadav et al. (2020).

It should be noted that such difference between languages could also be observed computing the length of dependency relations straightforwardly on PUD treebanks: the average linear link length computed on Hindi PUD is 6.54, for Thai PUD, the language showing shorter relations, is 2.67, while the remaining languages show a value ranging between 3.1 and 3.5. However, our methodology allows us to combine multiple properties simultaneously into a score, thus isolating in different portions of the rankings the *deprels* that show an atypical value for a given property but could be still considered quite typical for the language based on their context. As proof, observe that long and medium *deprels* in Hindi tend to appear earlier in the ranking than in other languages: 19.73% of deprels located in the middle bin are covered by medium and long *deprels*, suggesting that longer *deprels* are more common in Hindi. On the contrary, only 7% of *deprels* of the middle bin are long in Thai, pointing to their atypicality in the language.

The above results show the methodology's effectiveness for exploring tendencies and peculiarities of languages in multilingual studies. However, small samples like PUD treebanks are usually not suited for analysing infrequent phenomena (Taherdoost, 2016). Hence, one might wonder if we are actually capturing the atypicality of linguistic constructions, or instead, we are biased by phenomena underrepresented in the treebank. In the following Section, we will explore whether low LISCA scores might be associated with infrequent linguistic phenomena due to under-representation in the data used to build the SM.

[5] All the other languages are S(ubject)V(erb)O(bject) languages.

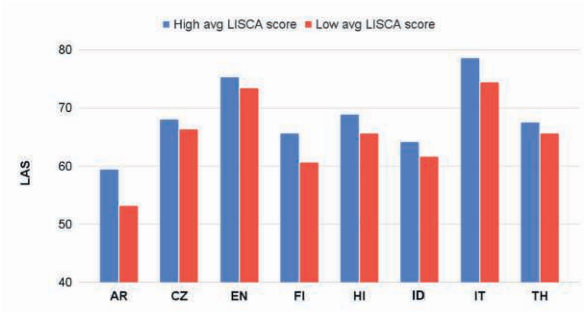


Figure 3: Parsing accuracy (LAS) on sentences having high and low LISCA scores.

4 Towards Treebank Expansion

Our analyses started from the premise that PUD treebanks are error-free. Therefore we can look at the rankings as containing correctly annotated examples of language use. However, the approach employed in this study does not exclude the scenario that a deprel might obtain a low LISCA score because of a lack of similar constructions in the treebank. We explored this idea both at *deprel* and sentence level, as described below.

Concerning the *deprel*–level analysis, we tested the accuracy of a parser for *deprels* in the three portions of the LISCA rankings. To this aim, we parsed each PUD treebanks using UDPipe (Straka et al., 2016), relying on the k-fold approach used to train LISCA: we split each PUD into 4 portions of 250 sentences each, trained UDPipe with $\frac{3}{4}$ of the portions and parsed the remaining portion. Then, we checked if *deprels* were parsed accurately. Again, we excluded function words from this analysis to improve cross-language comparability and avoid biased results as function words are usually more accurately parsed than content words. We observed that wrongly parsed deprels mainly concentrate in the bottom bins for all languages based on the obtained results. This suggests that there might be a relationship between low LISCA scores and underrepresented phenomena.

For the sentence-level analysis, we computed the LISCA score for each sentence in all PUD treebanks as the arithmetic mean of the scores of the individual *deprels* belonging to the sentence to get a sentence–level LISCA score. In the analysis, we explored whether sentences with low average LISCA scores are also more difficult to parse than those with higher average LISCA scores. Having computed the sentence–level LISCA scores, we collected two test sets of 100 sentences each by grouping sentences showing the highest and lowest LISCA scores. Then, we trained UDPipe using the remaining 800 sentences of PUD. The performances of UDPipe on the test sets are reported in terms of Labelled Attachment Score (LAS).

The results of this experiment are reported in Figure 3. We observe that the test sets composed of sentences characterised by the highest scores are more accurately parsed than the lower-score sets for all the languages involved. Differences between languages in terms of overall Label Attachment Score (LAS) and between the two subgroups of sentences will be further investigated in future work. Such results complement the *deprel*-level analysis: they suggest that the methodology could isolate difficult-to-parse sentences, and not only *deprels*, that could be employed to expand treebanks.

Treebank expansion is extremely valuable for low-resourced languages and small resources in general as it allows to include unseen examples to treebanks. Our results suggest that the sentence suites collected by grouping sentences characterised by the lowest LISCA scores contain difficult-to-parse constructions, possibly underrepresented in PUD, that should be included in the treebank to improve its representativeness.

5 Conclusion

We proposed a novel workflow to adapt an existing approach for treebank exploration to small treebanks and low-resourced languages. Results of our analyses showed the effectiveness of the methodology in multiple scenarios. First, the adapted method allows obtaining reliable results on par with the original method workflow when performing linguistic explorations of the treebanks. Secondly, the results also show the potential of the method for automatically identifying underrepresented constructions in treebanks. The latter result paves the way for the automatic identification of cases required to expand the treebanks, which we plan to further investigate in future work.

Acknowledgments

We would like to sincerely thank the anonymous reviewers for their helpful comments.

References

Akshay Aggarwal. 2020. Consistency of Linguistic Annotation. Master's thesis, Univerzita Karlova (ÚFAL), Prague, Czechia, September. Thesis Supervisor Zeman, Daniel.

Chiara Alzetta, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2017. Dangerous Relations in Dependency Treebanks. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 201–210, Prague, Czech Republic.

Chiara Alzetta, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2019a. Inferring quantitative typological trends from multilingual treebanks. A case study. *Lingue e linguaggio*, 18(2):209–242.

Chiara Alzetta, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2019b. Inferring quantitative typological trends from multilingual treebanks. A case study. *Lingue e Linguaggio*, XVIII(2):209–242.

Chiara Alzetta, Felice Dell'Orletta, Simonetta Montemagni, Petya Osenova, Kiril Simov, and Giulia Venturi. 2020a. Quantitative Linguistic Investigations across Universal Dependencies treebanks. In *Proceedings of the Seventh Italian Conference on Computational Linguistics (CLiC-it)*, Bologna (online), Italy, March.

Chiara Alzetta, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2020b. Linguistically-driven Selection of Difficult-to-Parse Dependency Structures. *IJCoL. Italian Journal of Computational Linguistics*, 6(6-2):37–60.

Mark Anderson, Anders Søgaard, and Carlos Gómez-Rodríguez. 2021. Replicating and Extending "Because Their Treebanks Leak": Graph Isomorphism, Covariants, and Parser Performance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 1090–1098.

Aleksandrs Berdicevskis, Çağrı Çöltekin, Katharina Ehret, Kilu von Prince, Daniel Ross, Bill Thompson, Chunxiao Yan, Vera Demberg, Gary Lupyan, Taraka Rama, et al. 2018. Using Universal Dependencies in cross-linguistic complexity research. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 8–17.

William Croft, Dawn Nordquist, Katherine Looney, and Michael Regan. 2017. Linguistic Typology meets Universal Dependencies. In *Proceedings of the 15th International Workshop on Treebanks and Linguistic Theories (TLT15)*, CEUR Workshop Proceedings, pages 63–75.

Marie-Catherine de Marneffe, Christopher D Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal dependencies. *Computational linguistics*, 47(2):255–308.

Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2013. Linguistically-driven Selection of Correct Arcs for Dependency Parsing. *Computación y Sistemas*, 17(2):125–136.

Vera Demberg and Frank Keller. 2008. Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 109(2):193–210.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Richard Futrell, Kyle Mahowald, and Edward Gibson. 2015. Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2021. A Survey on Recent Approaches for Natural Language Processing in Low-Resource Scenarios. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2545–2568, Online, June. Association for Computational Linguistics.

Jingyang Jiang and Haitao Liu. 2018. *Quantitative Analysis of Dependency Structures*, volume 72. Walter de Gruyter GmbH & Co KG.

Ryan McDonald and Joakim Nivre. 2011. Analyzing and integrating dependency parsers. *Computational Linguistics*, 37(1):197–230.

Joakim Nivre. 2015. Towards a universal grammar for natural language processing. In *International conference on intelligent text processing and computational linguistics*, pages 3–16. Springer.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Anders Søgaard. 2020. Some Languages Seem Easier to Parse Because Their Treebanks Leak. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2765–2770.

Milan Straka, Jan Hajic, and Jana Straková. 2016. UD-Pipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the tenth international conference on language resources and evaluation (LREC 2016)*, pages 4290–4297.

Hamed Taherdoost. 2016. Sampling methods in research methodology; how to choose a sampling technique for research. *How to Choose a Sampling Technique for Research (April 10, 2016)*.

David Temperley. 2007. Minimization of dependency length in written English. *Cognition*, 105(2):300–333.

Clara Vania, Yova Kementchedjhieva, Anders Søgaard, and Adam Lopez. 2019. A systematic comparison of methods for low-resource dependency parsing on genuinely low-resource languages. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1105–1116.

Ekaterina Vylomova, Edoardo M Ponti, Eitan Grossman, Arya D McCarthy, Yevgeni Berzak, Haim Dubossarsky, Ivan Vulić, Roi Reichart, Anna Korhonen, and Ryan Cotterell. 2020. Proceedings of the Second Workshop on Computational Research in Linguistic Typology. In *Proceedings of the Second Workshop on Computational Research in Linguistic Typology*.

Himanshu Yadav, Ashwini Vaidya, Vishakha Shukla, and Samar Husain. 2020. Word Order Typology Interacts With Linguistic Complexity: A Cross-Linguistic Corpus Study. *Cognitive science*, 44(4):e12822.

Xiang Yu, Agnieszka Falenska, and Jonas Kuhn. 2019. Dependency length minimization vs. word order constraints: an empirical study on 55 treebanks. In *Proceedings of the First Workshop on Quantitative Syntax (Quasy, SyntaxFest 2019)*, pages 89–97.

Daniel Zeman, Martin Popel, Milan Straka, Jan Hajic, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinkova, Jan Hajic jr., Jaroslava Hlavacova, Václava Kettnerová, Zdenka Uresova, Jenna Kanerva, Stina Ojala, Anna Missilä, Christopher D. Manning, Sebastian Schuster, Siva Reddy, Dima Taji, Nizar Habash, Herman Leung, Marie-Catherine de Marneffe, Manuela Sanguinetti, Maria Simi, Hiroshi Kanayama, Valeria de-Paiva, Kira Droganova, Héctor Martínez Alonso, Çağrı Çöltekin, Umut Sulubacak, Hans Uszkoreit, Vivien Macketanz, Aljoscha Burchardt, Kim Harris, Katrin Marheinecke, Georg Rehm, Tolga Kayadelen, Mohammed Attia, Ali Elkahky, Zhuoran Yu, Emily Pitler, Saran Lertpradit, Michael Mandl, Jesse Kirchner, Hector Fernandez Alcalde, Jana Strnadová, Esha Banerjee, Ruli Manurung, Antonio Stella, Atsuko Shimada, Sookyoung Kwak, Gustavo Mendonca, Tatiana Lando, Rattima Nitisaroj, and Josie Li. 2017. Conll 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada, August. Association for Computational Linguistics.

Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. CoNLL 2018 shared task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21, Brussels, Belgium, October. Association for Computational Linguistics.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Hamid Aghaei, Željko Agić, Amir Ahmadi, Lars Ahrenberg, Chika Kennedy Ajede, Gabrielė Aleksandravičiūtė, Ika Alfina, Lene Antonsen, Katya Aplonova, Angelina Aquino, Carolina Aragon, Maria Jesus Aranzabe, Bilge Nas Arıcan, Ĥórunn Arnardóttir, Gashaw Arutie, Jessica Naraiswari Arwidarasti, Masayuki Asahara, Deniz Baran Aslan, Luma Ateyah, Furkan Atmaca, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Keerthana Balasubramani, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Starkaður Barkarson, Victoria Basmov, Colin Batchelor, John Bauer, Seyyit Talha Bedir, Kepa Bengoetxea, Gözde Berk, Yevgeni Berzak, Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė, Kristín Bjarnadóttir, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Anouck Braggaar, Kristina Brokaitė, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Lauren Cassidy, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Neslihan Cesur, Savas Cetin, Özlem Çetinoğlu, Fabricio Chalub, Shweta Chauhan, Ethan Chi, Taishi Chika, Yongseok Cho, Jinho Choi, Jayeol Chun, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Mihaela Cristescu, Philemon. Daniel, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Mehmet Oguz Derin, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Arawinda Dinakaramani, Elisa Di Nuovo, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Hanne Eckhoff, Sandra Eiche, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Aline Etienne, Wograine Evelyn, Sidney Facundes, Richárd Farkas, Marília Fernanda, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Fabrício Ferraz Gerardi, Kim Gerdes, Filip Ginter, Gustavo Godoy, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Loïc Grobol, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Tunga Güngör, Nizar Habash, Hinrik Hafsteinsson, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-

Rae Han, Muhammad Yudistira Hanifmuti, Sam Hardwick, Kim Harris, Dag Haug, Johannes Heinecke, Oliver Hellwig, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Eva Huber, Jena Hwang, Takumi Ikeda, Anton Karl Ingason, Radu Ion, Elena Irimia, Ọlájídé Ishola, Kaoru Ito, Tomáš Jelínek, Apoorva Jha, Anders Johannsen, Hildur Jónsdóttir, Fredrik Jørgensen, Markus Juutinen, Sarveswaran K, Hüner Kaşıkara, Andre Kaasen, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Neslihan Kara, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Arne Köhn, Abdullatif Köksal, Kamil Kopacewicz, Timo Korkiakangas, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Parameswari Krishnamurthy, Oğuzhan Kuyrukçu, Aslı Kuzgun, Sookyoung Kwak, Veronika Laippala, Lucia Lam, Lorenzo Lambertino, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phương Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Maria Levina, Cheuk Ying Li, Josie Li, Keying Li, Yuan Li, KyungTae Lim, Bruna Lima Padovani, Krister Lindén, Nikola Ljubešić, Olga Loginova, Andry Luthfi, Mikko Luukko, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Büşra Marşan, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Hiroshi Matsuda, Yuji Matsumoto, Alessandro Mazzei, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Niko Miekka, Karina Mischenkova, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, AmirHossein Mojiri Foroushani, Judit Molnár, Amirsaeid Moloodi, Simonetta Montemagni, Amir More, Laura Moreno Romero, Giovanni Moretti, Keiko Sophie Mori, Shinsuke Mori, Tomohiko Morioka, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Mariam Nakhlé, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Manuela Nevaci, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Alireza Nourian, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Adédayọ Olúòkun, Mai Omura, Emeka Onwuegbuzia, Petya Osenova, Robert Östling, Lilja Øvrelid, Şaziye Betül Özateş, Merve Özçelik, Arzucan Özgür, Balkız Öztürk Başaran, Hyunji Hayley Park, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Angelika Peljak-Łapińska, Siyao Peng, Cenel-Augusto Perez, Natalia Perkova, Guy Perrier, Slav Petrov, Daria Petrova, Jason Phelan, Jussi Piitulainen, Tommi A Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andriela Rääbis, Alexandre Rademaker, Taraka Rama, Loganathan Ramasamy, Carlos Ramisch, Fam Rashel, Mohammad Sadegh Rasooli, Vinit Ravishankar, Livy Real, Petru Rebeja, Siva Reddy, Georg Rehm, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Eiríkur Rögnvaldsson, Mykhailo Romanenko, Rudolf Rosa, Valentin Roșca, Davide Rovati, Olga Rudina, Jack Rueter, Kristján Rúnarsson, Shoval Sadde, Pegah Safari, Benoît Sagot, Aleksi Sahala, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Ezgi Sanıyar, Dage Särg, Baiba Saulīte, Yanin Sawanakunanon, Shefali Saxena, Kevin Scannell, Salvatore Scarlata, Nathan Schneider, Sebastian Schuster, Lane Schwartz, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Yana Shishkina, Muh Shohibussirri, Dmitry Sichinava, Janine Siewert, Einar Freyr Sigurðsson, Aline Silveira, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Maria Skachedubova, Aaron Smith, Isabela Soares-Bastos, Carolyn Spadine, Rachele Sprugnoli, Steinh͡ór Steingrímsson, Antonio Stella, Milan Straka, Emmett Strickland, Jana Strnadová, Alane Suhr, Yogi Lesmana Sulestio, Umut Sulubacak, Shingo Suzuki, Zsolt Szántó, Dima Taji, Yuta Takahashi, Fabio Tamburini, Mary Ann C. Tan, Takaaki Tanaka, Samson Tella, Isabelle Tellier, Marinella Testori, Guillaume Thomas, Liisi Torga, Marsida Toska, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk, Francis Tyers, Sumire Uematsu, Roman Untilov, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Sowmya Vajjala, Rob van der Goot, Martine Vanhove, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Natalia Vlasova, Aya Wakasa, Joel C. Wallenberg, Lars Wallin, Abigail Walsh, Jing Xian Wang, Jonathan North Washington, Maximilan Wendt, Paul Widmer, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Kayo Yamashita, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Arife Betül Yenice, Olcay Taner Yıldız, Zhuoran Yu, Zdeněk Žabokrtský, Shorouq Zahra, Amir Zeldes, Hanzhi Zhu, Anna Zhuravleva, and Rayan Ziane. 2021. Universal dependencies 2.8.1. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

On the Role of Textual Connectives in Sentence Comprehension: A New Dataset for Italian

Giorgia Albertin[•], **Alessio Miaschi**[⋆ ◇], **Dominique Brunato**[◇]
[•] University of Padova
[⋆]Department of Computer Science, University of Pisa
[◇]Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC–CNR)
ItaliaNLP Lab - *www.italianlp.it*
giorgia.albertin.2@studenti.unipd.it,
alessio.miaschi@phd.unipi.it, dominique.brunato@ilc.cnr.it

Abstract

In this paper we present a new evaluation resource for Italian aimed at assessing the role of textual connectives in the comprehension of the meaning of a sentence. The resource is arranged in two sections (acceptability assessment and cloze test), each one corresponding to a distinct challenge task conceived to test how subtle modifications involving connectives in real usage sentences influence the perceived acceptability of the sentence by native speakers and Neural Language Models (NLMs). Although the main focus is the presentation of the dataset, we also provide some preliminary data comparing human judgments and NLMs performance in the two tasks[1].

1 Introduction

The outstanding performance reached by recent Neural Language Models (NLMs) across a variety of NLP tasks that require extensive linguistic skills has stimulated an increased interest in the theoretical and computational linguistics community towards a better understanding of their inner mechanisms. In particular, the debate is focused on trying to understand what kind of linguistic knowledge these models are able to induce from the raw data they are exposed to and to what extent this knowledge resembles human-like generalization patterns (Linzen and Baroni, 2021; Manning, 2015). To pursue this investigation, it has become of pivotal importance the availability of challenging test sets, also called 'diagnostic' or 'stress' tests, built to probe the sensitivity of a model to specific language phenomena.

So far, most of the efforts have been focused on assessing the syntactic abilities encoded by NLMs by exploiting human curated benchmarks, which are usually proposed in the form of minimal sentence pairs, i.e. minimally different sentences exemplifying a wide array of linguistic contrasts. A well-known one is BLiMP (Benchmark of Linguistic Minimal Pairs) (Warstadt et al., 2020) which contains pairs that contrast in syntactic acceptability and isolating fine-grained phenomena in specific domains of the English grammar, such as subject–verb agreement, island effects, ellipsis and negative polarity items.

Differently from syntactic well-formedness, less explored is the sensitivity of these models to deeper linguistic dimensions involving semantics and discourse, such as textual cohesion, which are critical to language understanding. With this respect, one of the explicit devices that natural languages use to convey textual cohesion is represented by function words. As observed by Kim et al. (2019), although these words play a key role in compositional meaning as they introduce discourse referents or make explicit relations between them, they are still under-investigated in the literature on representation learning. To this end, the authors released a suite of nine challenge tasks for English aimed to test the NLMs' understanding of specific types of function word, e.g. coordinating conjunctions, quantifiers, definite articles. Reasoning about conjuncts in conjunctive sentences, Saha et al. (2020), instead, introduced CONJNLI, a challenge stress-test for Natural Language Inference (NLI) over conjunctive sentences, where the premise differs from the hypothesis by conjuncts removed, added, or replaced.

Taking inspiration from this work, in this paper we focus the attention on the role of textual connectives in the comprehension of a sentence and we introduce a new evaluation resource for Italian which, to our knowledge, is the first one for this language. The resource is articulated into two sections

[1]The resource is available at: http://www.italianlp.it/resources/.

(acceptability assessment and cloze test), each one corresponding to a distinct task aimed at probing, in a different format, to what extent current NLMs are able to properly encode the role of connectives in a sentence. A peculiarity of the dataset is that it contains sentences that were extracted and minimally modified from existing corpora so as to test the comprehension of connectives in the real use of language.

2 Corpus Collection

This section is divided into two parts. In the first one, we discuss the methodology implemented for the selection of connectives and the extraction of the sentences. Subsequently, we provide an overview of the two tasks defined to test the correct comprehension of connectives.

2.1 Selecting Connectives and Extracting Sentences

As a first step, we defined the linguistic criteria for the selection of connectives to include in the corpus. By *connective* we mean specific words that have the function of drawing a relation between two or more clauses (Sanders and Noordman, 2000; Graesser and McNamara, 2011). To this end, two resources were employed: the INVALSI reading comprehension and language reflection tests designed by the National Institute for the Evaluation of the Education System and the *Nuovo Vocabolario di Base* of Italian (NVdB) (De Mauro and Chiari, 2016). Starting from the collection of the INVALSI tests proposed in the last six years for different grades, we extracted all words which were expressly called 'connective' in the tests or were involved in defining a logical relationship between two sentences. We thus obtained a first list of 46 elements, belonging to diverse morpho-syntactic categories (i.e. prepositions, conjunctions, adverbs), which was then integrated with other 19 connectives extracted from the *NVdB*. We then checked the distribution of the selected items in existing Italian treebanks and extracted the sentences in which these words were unambiguously used as sentence connectives. Three different sections of the Italian Universal Dependency Treebank (IUDT) (Zeman et al., 2020) were used: ISDT (Bosco et al., 2013), PoSTWITA (Sanguinetti et al., 2018) and TWITTIRò (Cignarella et al., 2019)[2], the first one representative of standard language and the latter collecting Italian tweets. We employed PML TreeQuery[3] to query the treebanks and filter the sentences containing the connectives we were interested in. In particular, to exclude occurrences which do not have the role of phrasal connectives (e.g. the conjunction *e* joining two nouns), only sentences in which the connective was headed by a verb or a copula were taken into account. We observed that the absolute frequency's positions of the selected connectives in the three corpora above-mentioned mostly overlap, although their occurrences in PoSTWITA and TWITTIRò (jointly considered as sample of Italian social media language) were lower than in ISDT, also given the different corpora sizes (i.e. 289,343 words in ISDT vs 154,050 words in PoSTWITA and TWITTIRò). Given the partial overlapping of the frequency data and the potential non-standard use of connectives in treebanks representative of social media texts, also due to genre-specific features (e.g. hashtag, emoticons etc.), we decided to consider only the first 21 most frequent connectives occurring in ISDT. As the first Italian corpus for the comprehension on textual connectives, we prefer to focus in sentences as close as possible to standard Italian language. Further considerations on connectives' distributions led us to the deletion of *per*, *così*, *ancora*, because of their ambiguous behavior as textual connectives (e.g. we noticed that the majority of the occurrences of *per* involves the presence of an infinite verb, a distribution which is far from the other connectives). The following 18 connectives were finally considered: *e*, *se*, *quando*, *come*, *ma*, *dove*, *o*, *anche*, *perché*, *poi*, *mentre*, *infatti*, *prima*, *però*, *invece*, *inoltre*, *tuttavia*, *quindi*. The distribution of the finally selected connectives from ISDT and from PoSTWITA and TWITTIRò is reported in Appendix A.

Once established the final list, those sentences which we consider more suitable to be involved in our tasks were manually extracted from ISDT and eventually modified following some patterns, to guarantee sentence comprehension. For example, in some cases two sentences occurring in the treebank in a subsequent order, but that were clearly extracted from the same text, were joined together to form a unique sentence, through the insertion of the appropriate punctuation. This happened e.g. when the connective appeared at the beginning of the second sentence joining this to the first one,

[2] `https://universaldependencies.org/treebanks/it-comparison.html`.

[3] `https://ufal.mff.cuni.cz/pmltq`.

which serves as the antecedent to comprehend the logical relationship. We tried to include in the dataset sentences with different degrees of syntactic and lexical complexity, considering the number of subordinate clauses and the variety of the lexicon as related proxies. All the original sentences, later arranged into the acceptability assessment and the cloze test task, are drawn from ISDT.

2.2 Definition of the Tasks

The collected sentences were grouped in two sections aimed at testing the correct comprehension of connectives in a different format, i.e. through an acceptability assessment task and a cloze test task. Table 1 provides an example of sentences/sentences pairs for each task.

2.2.1 Acceptability Assessment Section

To design the acceptability assessment task, we selected 15 sentences per connective from the whole dataset. For each sentence, an unacceptable counterpart was created by replacing the original connective with another of the list. The replacement strategy was meant to obtain unacceptable sentences with contradictory or nonsensical meaning but preserving their grammaticality. Indeed those sentences should be the most challenging one for NLMs, which have been shown to be capable of detecting sentence grammaticality (Jawahar et al., 2019), but still struggle to track down unacceptable meanings and contradictions. Nevertheless, we were not always able to guarantee this constraint as for some specific contexts none of the available connective could be substituted without affecting the resulting grammaticality. This happened in 98 cases, which we decided to keep in the dataset but we signaled with the label ‘no’ in the field ’grammaticality’, as in:

> *Nei campi si sopravvive* anche *intorno tutto muore.*

Although the assessment of grammaticality is not the main focus of this work, given the fact that it was unavoidably violated in the above-reported cases, we feel compelled to provide distinguished analysis for the group of ungrammatical sentences. A few sentences were also deleted due to ambiguity. The final section contains 518 sentence pairs, i.e. 259 acceptable and 259 unacceptable ones.

2.2.2 Cloze Test Section

The second section was designed as a cloze test task and contains 270 sentences, 15 for connective. For every sentence the original connective was replaced by a blank space and 5 alternatives were proposed for completion: the target, a plausible alternative and three implausible options. For ‘plausible alternative’ we mean another connective of the list that could occupy the same linguistic contest of the target, yielding to an identical meaning or to a different, yet totally plausible, reading. As for the acceptability task, it turns out that for some connectives (e.g. *prima*) it was very challenging, if not impossible, to propose such a plausible connective. In those cases, that in truth are only a minority, it has been proposed an alternative that at least should guarantee the grammaticality.

3 Corpus Annotation

The two sections of the dataset were splitted into 9 surveys (5 for the acceptability assessment task and 4 for the cloze task) and submitted to human evaluation by recruiting Italian native speakers of different ages through the Prolific platform[4].

In the **acceptability assessment task**, participants were asked to judge the acceptability of each sentence on a 5-grade Likert scale (from 1=‘totally unacceptable’ to 5=‘totally acceptable’). Although this makes the dataset more challenging, we assume that acceptability is a gradual rather than binary notion as it is affected by many factors (Sorace and Keller, 2005; Sprouse, 2007). To disambiguate the interpretation of sentence acceptability and orient annotators in giving their judgments, the survey guidelines encouraged them to think if they found the sentence natural in Italian and if they would have used it in a real conversation or any other communicative context.

For the **cloze test task**, participants were required to supply the missing element choosing among the proposed options plus the one “none of the previous options is suitable”.

Each survey was completed by 20 annotators on average. The number of annotations per sentence in the acceptability task ranges from 16 to 21 and for the cloze task from 18 to 21. To improve data quality, we discarded annotators who took less than 10 minutes to complete the test, considering the average threshold time for each survey. This led us to reject 5 annotators only for the acceptability task.

Table 2 reports the average human score and standard deviation obtained by the acceptable and

[4] `https://prolific.co`.

Section	Id	Sentence
Acceptability	e_11A	L'arte e la scienza sono libere e libero ne è l'insegnamento.
	e_11NA	L'arte e la scienza sono libere tuttavia libero ne è l'insegnamento.
	ma_64A	Paolo si muove con difficoltà, ma è sempre allegro e di buon umore.
	ma_64NA	Paolo si muove con difficoltà, perché è sempre allegro e di buon umore
Cloze test	se_23cl	Che cosa possiamo fare in estate ... vogliamo partire per le vacanze e abbiamo un cane o un gatto? [**se** *quando* perché dove come]
	mentre_162cl	Nelle botteghe artigianali della produzione di piastrelle la smaltatura è ancora tradizionale, ... i forni, come è naturale, oggi funzionano a gas. [**mentre** *invece* come dove perché]

Table 1: Examples from the dataset. Sentences are indicated with the last part of id, which gives information about the target connective, the position of the sentence in the section and the label in each section (A='acceptable', NA='not acceptable'; cl='cloze test'). For the cloze task, the target connective is marked in bold and the plausible alternative in italics.

Acceptability label	AvgIntScore	(StDev)
Acceptable	4.286	0.519
NonAcceptable	1.822	0.451
NonAccept+NonGr	1.616	0.350

Table 2: Average scores assigned by humans (with standard deviation) to the acceptable, unacceptable and unacceptable+ungrammatical sentences.

Cloze task choice	N. Items	(%)
Target	213	78.89
Plausible alt.	48	17.78
Implausible alt.	4	1.48
Target=Plausible alt.	5	1.85

Table 3: Number and % of sentences for which the majority label was assigned to the target connective, to the plausible alternative, to an implausible alternative or equally balanced between the target and the plausible alternative.

unacceptable sentences. For the latter, we separately computed these scores for the subset of sentences which were also labeled as ungrammatical (see Section 2.2.1). As it can be seen, humans perform very well on the task assigning quite higher scores to the acceptable sentences with respect to the unacceptable ones, also with little variability. Within the unacceptable subset, the slightly smaller score received on average by ungrammatical sentences provides further evidence that humans are sensitive to this distinction.

Also for the **cloze test task** the human evaluation confirms the validity of the resource. Indeed, as shown in Table 3, the target connective was largely chosen by the majority of annotators as the most adequate one, although for ∼20% of sentences humans preferred the plausible candidate or the two options got half annotations each. The percentage of sentences for which the majority label was given to an implausible choice is largely negligible.

4 Testing the Sensitivity of Neural Language Models to Connectives

We conclude by presenting some preliminary findings aimed at testing the performance of NLMs in the two tasks. Specifically, we performed two distinct evaluations. For the **acceptability assessment task**, we computed the perplexity (*PPL*) score assigned by the GePpeTto model (De Mattei et al., 2020) to all sentences of the corresponding section. We relied on perplexity as it is a standard evaluation measure of the quality of a language model yielding a good approximation of how well a model recognises an unseen piece of text as a plausible one. Accordingly, we assumed that higher *PPL* scores should be assigned to sentences labeled as unacceptable with respect to their original version. GePpeTto was chosen as it is a traditional unidirectional model built using the GPT-2 architecture (Radford et al., 2019) and, differently from a bidirectional model such as BERT (Devlin et al., 2019), allows computing a well-formed probability distribution over sentences. The sentence-level *PPL* was calculated using the formula reported in Miaschi et al. (2020).

By inspecting the results in Table 4, we observed that the average *PPL* score assigned to the acceptable sentences is quite lower than the one assigned to the unacceptable ones (i.e. 42.512 vs 78.280).

As expected, for the subset of unacceptable sentences, perplexity was on average higher for the ones marked as ungrammatical (98.992), reflecting

AcceptabilityLabel	AvgPPL	minPPL	maxPPL
Acceptable	42.512	2.059	455.961
NonAcceptable	78.280	3.534	390.824
NonAccept+NonGr	98.992	9.933	1178.162

Table 4: Average, minimum and maximum perplexity value given by the model to the acceptable, unacceptable and unacceptable+ungrammatical sentences.

the model's capability of encoding syntactic phenomena. Interestingly, among unacceptable sentences, those obtaining lower *PPL* scores were perfectly well-formed but with an implausible meaning, as in the case of:

> *Il film 'Le chiavi di casa' ha partecipato al Festival del Cinema di Venezia di quest'anno,* ***perché*** *non ha vinto nessun premio* ($PPL =$ 13.892).

To compare humans and model performance, we also computed the Spearman's rank correlation (ρ) between the average acceptability score given by annotators and the *PPL* score assigned by the model to the same sentences. Although limited to this analysis, the resulting very weak correlation (i.e. $\rho = -0.120$, $p - value < 0.01$) suggests that connectives differently impact on the ability of humans and models to assess the plausibility of a sentence.

As for the **cloze task** test, we relied on the pretrained Italian version of the BERT model developed by the MDZ Digital Library Team and available trough the Huggingface's *Transformers* library (Wolf et al., 2020) [5]. We extracted the first ten completions provided by the model trough the Masked Language Modeling task (MLM) for each sentence, along with their probabilities. This allowed us to inspect whether and in how many cases either the target connective or the plausible alternative appear in the top-ranked predictions.

As shown in Table 5, for the large majority of cases BERT is able to infer in its first 10 predictions that the sentence should be completed with a correct connective. That happens in 86.29% of the sentences for the target, resulting from the sum of the cases where only the target occurs in the completions (31.48%) with the cases in which both the target and the plausible alternative were predicted (54.81%), and in 59.25% for the plausible alternative (that is 4.44% plus 54.81%). Focusing instead on the first completion for each sentence, we observe that in almost half of the sentences BERT assigns the highest probability to the original connective (41.11%) or to the plausible one (8.52%).

Predict.	10_match		1st_match	
Target	(85)	31.48%	(111)	41.11%
Pl. alt.	(12)	4.44%	(23)	8.52%
Target+Pl. alt.	(148)	54.81%	–	–
Other	(25)	9.26%	–	–

Table 5: (Number) and % of BERT's completions in which only the target, only the plausible alternative, both of them or none of them (*Other*) occur in the first 10 predictions (*10_match*). (Number) and % of the completions in which the target and the plausible alternative were predicted with the highest probability are also reported (*1st_match*).

We are currently performing a more qualitative analysis to better investigate the cases in which the correct connective hasn't received a high probability score, as well as those in which neither of the two options appeared at all (i.e. *Other* cases in Table 5), in order to understand whether the other completions can still be considered as plausible ones. Preliminary findings showed that, among the *Other* cases, about 56 of the completions provided by BERT are unacceptable and 34 of them are dubious acceptable i.e. not clearly recognizable as acceptable[6], as in the case of the following sentence[7]:

> *Secondo gli esperti, in Italia i giovani leggono meno i giornali rispetto ai giovani di altri Paesi europei, ... rispetto agli anni passati i giovani tra i 14 e i 19 anni leggono più spesso i giornali.* [**perché** anche *però*].

Nevertheless, the majority of *Other*'s completions can be considered as acceptable ones. In fact, BERT predicted a word leading to the same meaning (or, at least, very similar) to the original sentence in more that 60 cases. Moreover, in most cases (i.e. 92) the completions provided are plausible ones, although in some of them the sentences acquire different meanings.

[5] `https://huggingface.co/dbmdz/bert-base-italian-xxl-cased`

[6] Note that in order to assign the acceptability label of each completion we refer to the usage of the Italian language as standard as possible.

[7] the unacceptable completion is marked in bold, the dubious acceptable one is reported in block and the original connective is indicated in italics.

5 Conclusion

In the context of studies devoted to assess the linguistic knowledge implicitly encoded by Neural Language Models, we introduced a new evaluation dataset for Italian designed to test the understanding of textual connectives in real-usage sentences. At first, we verified the significance of a set of selected connectives through a frequency analysis on already existing Italian gold corpora. Then, we manually selected only those sentences in which occur a genuine connective. Finally, we grouped the sentences into two different tasks, differing for the format used to elicit sentence comprehension in humans and current state-of-the-art NLMs: acceptability assessment and cloze test tasks. Human evaluation was provided for both the section, to verify the robustness of the dataset, which indeed was confirmed from the judgements collected.

Preliminary findings on NLMs behaviour on textual connectives showed that in several cases the models are capable of distinguishing between acceptable and unacceptable sentences, thus suggesting their ability to encode sentence meaning within their internal mechanisms. However, it remains unclear to what extent these models rely on semantic acceptability features, since we observed cases in which they fail to recognize implausible meaning of perfectly grammatical sentences.

We are currently increasing the dataset with the introduction of a new section designed in the form of the traditional Natural Language Inference task, for which the understanding of a given connective will be fundamental to infer the correct entailment relation between a premise and a hypothesis. We also believe that expanding the dataset to further connectives and including sentences representative of non standard italian language usage, i.e. social-media language, would be desirable to improve the robustness of the resource.

References

Cristina Bosco, Simonetta Montemagni, and Maria Simi. 2013. Converting italian treebanks: Towards an italian stanford dependency treebank. In *Proceedings of the ACL Linguistic Annotation Workshop & Interoperability with Discourse*.

Alessandra Teresa Cignarella, Cristina Bosco, and Paolo Rosso. 2019. Presenting TWITTIRÒ-UD: An italian twitter treebank in universal dependencies. In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*.

Lorenzo De Mattei, Michele Cafagna, Felice Dell'Orletta, Malvina Nissim, and Marco Guerini. 2020. Geppetto carves italian into a language model. In *CLiC-it*.

Tullio De Mauro and I Chiari. 2016. Il nuovo vocabolario di base della lingua italiana. *Internazionale.[28/11/2020]. https://www. internazionale. it/opinione/tullio-de-mauro/2016/12/23/il-nuovo-vocabolario-di-base-della-lingua-italiana*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Arthur C Graesser and Danielle S McNamara. 2011. Computational analyses of multilevel discourse comprehension. *Topics in cognitive science*, 3(2):371–398.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July. Association for Computational Linguistics.

Najoung Kim, Roma Patel, Adam Poliak, Alex Wang, Patrick Xia, R. Thomas McCoy, Ian Tenney, Alexis Ross, Tal Linzen, Benjamin Van Durme, Samuel R. Bowman, and Ellie Pavlick. 2019. Probing what different nlp tasks teach machines about function word comprehension. In **SEMEVAL*.

Tal Linzen and Marco Baroni. 2021. Syntactic structure from deep learning. *Annual Review of Linguistics*, 7:195–212.

Christopher D. Manning. 2015. Computational Linguistics and Deep Learning. *Computational Linguistics*, 41(4):701–707, 12.

Alessio Miaschi, Chiara Alzetta, Dominique Brunato, Felice Dell'Orletta, and Giulia Venturi. 2020. Is neural language model perplexity related to readability? In *CLiC-it*.

Alec Radford, Jeff Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Swarnadeep Saha, Yixin Nie, and Mohit Bansal. 2020. ConjNLI: Natural language inference over conjunctive sentences. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8240–8252, Online, November. Association for Computational Linguistics.

Ted JM Sanders and Leo GM Noordman. 2000. The role of coherence relations and their linguistic markers in text processing. *Discourse processes*, 29(1):37–60.

Manuela Sanguinetti, Cristina Bosco, Alberto Lavelli, Alessandro Mazzei, and Fabio Tamburini. 2018. PoSTWITA-UD: an Italian Twitter Treebank in universal dependencies. In *Proceedings of the Eleventh Language Resources and Evaluation Conference (LREC 2018)*.

Antonella Sorace and Frank Keller. 2005. Gradience in linguistic data. *Lingua*, 115(11):1497–1524.

Jon Sprouse. 2007. Continuous acceptability, categorical grammaticality, and experimental synta. *Biolinguistics*, pages 1123–134.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October. Association for Computational Linguistics.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Hamid Aghaei, Željko Agić, Amir Ahmadi, Lars Ahrenberg, Chika Kennedy Ajede, Gabrielė Aleksandravičiūtė, Ika Alfina, Lene Antonsen, Katya Aplonova, Angelina Aquino, Carolina Aragon, Maria Jesus Aranzabe, Ĥórunn Arnardóttir, Gashaw Arutie, Jessica Naraiswari Arwidarasti, Masayuki Asahara, Luma Ateyah, Furkan Atmaca, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Keerthana Balasubramani, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Victoria Basmov, Colin Batchelor, John Bauer, Seyyit Talha Bedir, Kepa Bengoetxea, Gözde Berk, Yevgeni Berzak, Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė, Kristín Bjarnadóttir, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Kristina Brokaitė, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Savas Cetin, Özlem Çetinoğlu, Fabricio Chalub, Ethan Chi, Yongseok Cho, Jinho Choi, Jayeol Chun, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Mehmet Oguz Derin, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Arawinda Dinakaramani, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Hanne Eckhoff, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Aline Etienne, Wograine Evelyn, Sidney Facundes, Richárd Farkas, Marília Fernanda, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Fabrício Ferraz Gerardi, Kim Gerdes, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Loïc Grobol, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Tunga Güngör, Nizar Habash, Hinrik Hafsteinsson, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Muhammad Yudistira Hanifmuti, Sam Hardwick, Kim Harris, Dag Haug, Johannes Heinecke, Oliver Hellwig, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Eva Huber, Jena Hwang, Takumi Ikeda, Anton Karl Ingason, Radu Ion, Elena Irimia, Ọlájídé Ishola, Tomáš Jelínek, Anders Johannsen, Hildur Jónsdóttir, Fredrik Jørgensen, Markus Juutinen, Sarveswaran K, Hüner Kaşıkara, Andre Kaasen, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Arne Köhn, Abdullatif Köksal, Kamil Kopacewicz, Timo Korkiakangas, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Parameswari Krishnamurthy, Sookyoung Kwak, Veronika Laippala, Lucia Lam, Lorenzo Lambertino, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phng Lê H`ông, Alessandro Lenci, Saran Lertpradit, Herman Leung, Maria Levina, Cheuk Ying Li, Josie Li, Keying Li, Yuan Li, KyungTae Lim, Krister Lindén, Nikola Ljubešić, Olga Loginova, Andry Luthfi, Mikko Luukko, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Hiroshi Matsuda, Yuji Matsumoto, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Niko Miekka, Karina Mischenkova, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, AmirHossein Mojiri Foroushani, Amirsaeid Moloodi, Simonetta Montemagni, Amir More, Laura Moreno Romero, Keiko Sophie Mori, Shinsuke Mori, Tomohiko Morioka, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Mariam Nakhlé, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Lng Nguy˜ên Thị, Huy`ên Nguy˜ên Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Alireza Nourian, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Adédayọ Olúòkun, Mai Omura, Emeka Onwuegbuzia, Petya Osenova, Robert Östling, Lilja Øvrelid, Şaziye Betül Özateş, Arzucan Özgür, Balkız Öztürk Başaran, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Angelika Peljak-Łapińska, Siyao Peng, Cenel-Augusto Perez, Natalia Perkova, Guy Perrier, Slav Petrov, Daria Petrova, Jason Phelan, Jussi Piitulainen, Tommi A Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andriela Rääbis, Alexandre Rademaker, Taraka Rama, Loganathan Ramasamy, Carlos Ramisch, Fam Rashel, Mohammad Sadegh Rasooli, Vinit Ravishankar, Livy Real, Petru Rebeja, Siva Reddy, Georg Rehm, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Eiríkur Rögnvaldsson, Mykhailo Romanenko, Rudolf Rosa, Valentin Roșca, Davide Rovati, Olga Rudina, Jack Rueter, Kristján Rúnarsson, Shoval Sadde, Pegah Safari, Benoît Sagot, Aleksi Sahala, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Dage Särg, Baiba Saulīte, Yanin Sawanakunanon, Kevin Scannell, Salvatore Scarlata, Nathan Schneider, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Muh Shohibussirri, Dmitry Sichinava, Einar Freyr Sigursson, Aline Silveira, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Maria Skachedubova, Aaron Smith, Isabela Soares-Bastos, Carolyn Spadine, Steinhór Steingrímsson, Antonio Stella, Milan Straka, Emmett Strickland, Jana Strnadová, Alane Suhr, Yogi Lesmana Sulestio, Umut Sulubacak, Shingo Suzuki, Zsolt Szántó, Dima Taji, Yuta Takahashi, Fabio Tamburini, Mary Ann C. Tan, Takaaki Tanaka, Samson Tella, Isabelle Tellier, Guillaume Thomas, Liisi Torga, Marsida Toska, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk, Francis Tyers, Sumire Uematsu, Roman Untilov, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Sowmya Vajjala, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Aya Wakasa, Joel C. Wallenberg, Lars Wallin, Abigail Walsh, Jing Xian Wang, Jonathan North Washington, Maximilan Wendt, Paul Widmer, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Kayo Yamashita, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Zhuoran Yu, Zdeněk Žabokrtský, Shorouq Zahra, Amir Zeldes, Hanzhi Zhu, and Anna Zhuravleva. 2020. Universal dependencies 2.7. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

A Appendix A

Conn.	**ISDT**		**PoSTWITA+TWITTIRò**	
e	(1,906)	0.639%	(909)	0.590%
se	(575)	0.193%	(477)	0.309%
quando	(529)	0.177%	(141)	0.092%
come	(422)	0.141%	(226)	0.147%
ma	(312)	0.105%	(713)	0.463%
dove	(306)	0.103%	(60)	0.039%
o	(259)	0.087%	(89)	0.058%
anche	(253)	0.085%	(123)	0.080%
perché	(231)	0.077%	(255)	0.166%
poi	(138)	0.046%	(46)	0.030%
mentre	(126)	0.042%	(24)	0.016%
infatti	(109)	0.037%	(13)	0.008%
prima	(106)	0.036%	(49)	0.032%
però	(101)	0.034%	(46)	0.030%
invece	(98)	0.033%	(49)	0.032%
inoltre	(88)	0.029%	(1)	0.0006%
tuttavia	(80)	0.027%	(1)	0.0006%
quindi	(78)	0.026%	(28)	0.018%

Table 6: (Numbers) and % of the frequency of the 18 finally selected connectives in ISDT corpus and in PoSTWITA+TWITTIRò corpora.

Query in linguaggio naturale per il dominio della dieta mediterranea

Luca Anselma, Dario Ferrero, Alessandro Mazzei
Dipartimento di Informatica, Università di Torino, Italy
{anselma,mazzei}@di.unito.it, dario.ferrero@edu.unito.it

Abstract

English. This paper presents an ongoing work for allowing users to ask questions in natural language to a database management system on the domain of recipes. The system translates questions from Italian language to SQL exploiting an interlingua represented by a logical formalism such as relational calculus. There are two specific features of this project: first, the use of relational calculus as semantic representation for interlingua translation; second, the role played by pragmatic information, that is the Mediterranean diet domain implicitly encoded in the database schema.

Italiano. *Questo articolo presenta un progetto in corso sulla per permettere a degli utenti di porre domande a un database management system nell'ambito del dominio delle ricette. Il sistema traduce le domande da italiano a SQL sfruttando una rappresentazione logica interlingua che consiste nel calcolo relazionale. Ci sono due specifiche caratteristiche di questo progetto: l'uso del calcolo relazionale come rappresentazione semantica della traduzione interlingua e il ruolo giocato dall'informazione pragmatica, che consiste nell'uso del dominio della dieta mediterranea che è stato codificata nello schema del database.*

1 Introduzione

L'accesso a grandi moli di dati da parte di persone comuni è una delle molteplici sfide affrontata da anni per permettere una più fluida interazione uomo-macchina. La natura della comunicazione umana presenta molti aspetti difficili da formalizzare tramite regole precise: ambiguità nell'uso di parole ed espressioni, differenze culturali ed idiomatiche, l'affidamento ad informazioni implicite date dal contesto. Questo è vero anche in specifici domini, come quello che lega le persone al cibo (Jurafsky, 2014). A questo scopo da diversi anni sono state argomento di ricerca le *Interfacce al Linguaggio Naturale (NLIs)*, sistemi in grado di interpretare, modellare ed eseguire un'interrogazione formulata in Linguaggio Naturale su un qualche tipo di base di conoscenza (basi di dati, ontologie, ecc.). Le applicazioni sono dunque tra le più svariate, dalle semplici barre di ricerca di un sito web, ai più recenti Voice Assistant ed altri servizi anche web-based (Balloccu et al., 2021). L'uso del linguaggio naturale ha vantaggi e svantaggi se paragonato a linguaggi formali come le query scritte in uno dei linguaggi di interrogazione dei database. Uno dei vantaggi è la familiarità dell'utente con il linguaggio naturale. Uno svantaggio consiste nell'estrema espressività del linguaggio umano, che rende impossibile una copertura totale del lessico e quindi delle possibili interpretazioni del significato.

In questo lavoro presentiamo una NLI per interrogare una base di dati nel dominio della dieta mediterranea. Una domanda su tale dominio espressa in lingua italiana verrà analizzata mediante l'uso di una grammatica *feature-based context-free*, rappresentata mediante il *calcolo relazionale su tuple*, e infine trasformata in *SQL* mediante un processo ricorsivo basato sulla pragmatica del dominio applicativo. Un elemento distintivo di questo progetto, rispetto ai classici esempi di conversione basati sulla logica del primo ordine (Warren and Pereira, 1982), è l'uso esplicito del *calcolo relazionale* per rappresentare il significato della frase. Inoltre, un ruolo importante è giocato dalla pragmatica del dominio, che lega il linguaggio usato nelle domande alla formalizzazione delle ricette, e che risulta codificata nello schema del database.

2 Alcuni lavori recenti LN→SQL basati sulla sintassi

In questa sezione consideriamo alcuni lavori sul tema della traduzione da linguaggio naturale a SQL che fanno uso della sintassi, seguendo lo schema proposto in (Affolter et al., 2019). Una rassegna più estesa si può trovare in (Amer-Yahia et al., 2021).

Negli *approcci basati sul Parsing* si derivano gran parte delle informazioni ricavabili a partire dalla struttura della frase in input. Tramite questo approccio è possibile riconoscere dipendenze complesse tra i singoli elementi sintattici, spesso introdotte da particolari espressioni o da preposizioni: si attribuisce quindi un valore semantico alla struttura logica della domanda analizzata. *Querix*, ad esempio, fornisce un'interfaccia per l'interrogazione di ontologie tramite la generazione di una query SPARQL (Kaufmann et al., 2006). Questo risultato è ottenuto tramite tre componenti: *query analyzer*, *matching center* e *query generator*. Il query analyzer parte dalla domanda in Linguaggio Naturale e, tramite componenti esterne quali lo Stanford Parser e il database lessicale WordNet, restituisce uno scheletro della query dato dalla sequenza delle categorie sintattiche proprie delle parole in input (Nome, Verbo, Preposizione, Congiunzione, ecc.). Tramite il matching center si cercano inizialmente dei pattern Soggetto-Proprietà-Oggetto all'interno delle categorie estratte dalla frase (la sua struttura assume quindi più importanza rispetto ai sistemi basati su keyword). Si cercano poi possibili match tra le parole in input e le risorse dell'ontologia (entrambi accresciuti dei rispettivi sinonimi di WordNet), ed infine si cerca di ottenere corrispondenze tra i risultati di queste due fasi intermedie. I matching delle triple così ottenute permettono al query generator di costruire una o più query SPARQL, ad ognuna delle quali è assegnato un ranking. L'ambiguità data dai molteplici risultati è risolta mostrando all'utente una finestra di dialogo che permette di selezionare la query più pertinente. L'efficacia del sistema dipende in gran parte dalla qualità del vocabolario dato dall'ontologia su cui è usato, limitando potenzialmente il range di domande esprimibili. Questa assenza di adattabilità è sia la debolezza che il maggior punto di forza di Querix, in quanto garantisce la più completa portabilità sui domini di qualsiasi ontologia utilizzata.

Gli *approcci basati su Grammatiche* si appoggiano largamente sulle regole di produzione: a differenza dei sistemi descritti in precedenza, in questi approcci è possibile stabilire a priori quali tipi di domande sia interpretabile e quali no. Da questa capacità segue quindi la possibilità di implementare un interfacciamento più interattivo, dove la validità di una domanda in input può essere determinata durante la stessa fase di inserimento e guidata da parte di feedback visivi. In *TR Discover*, ad esempio, la prima fase di traduzione della query consiste nel suo parsing su una *Feature-Based Context-Free Grammar* (FCFG) dove regole lessicali, generate dai nomi degli attributi della base di dati, e regole grammaticali, le quali definiscono la composizione delle informazioni estratte, permettono di ottenere composizionalmente come rappresentazione intermedia una formula nella Logica del Primo Ordine (FOL) (Song et al., 2015). Generare questo primo tipo di formulazione porta una notevole flessibilità permettendo una traduzione finale sia in SQL che in SPARQL. Nella seconda fase è effettuata un'ulteriore fase di analisi, questa volta sulla formula logica, tramite un FOL parser e un'apposita grammatica. Un importante punto di forza di TR Discover consiste nel modulo di *auto-suggestion*: il fatto che la grammatica definisca precisamente il tipo di domande valide può essere sfruttato dal sistema per suggerire all'utente possibili domande. A partire quindi dalle parole di un input parziale, i suggerimenti sono generati dalle strutture grammaticali derivabili e ordinati a seconda della "popolarità" delle entità riconosciute. Il sistema, tuttavia, presenta chiare limitazioni di espressività, non essendo in grado di analizzare query richiedenti quantificazioni e permettendo le negazioni solo traducendo in SPARQL.

3 Dominio applicativo: il progetto MAdiMan

Nel dominio della dieta alimentare il progetto *MADiMAN (Multimedia Application for Diet MANagement)* si pone come obiettivo quello di fornire un intermediario semi-automatizzato (un "dietista virtuale") tra l'utente e la gestione della sua alimentazione: a partire da informazioni quali dati biologici individuali, una definizione della dieta da seguire (basata su quantità di macronutrienti) e un aggiornamento costante sullo storico dei pasti, il sistema cloud-based è in grado di proporre diversi servizi volti alla pianificazione dei pasti futuri, alla persuasione dell'utente verso una sana alimentazione e all'interazione efficace con queste informazioni (Anselma et al., 2018; Anselma and Mazzei, 2020; Mazzei et al., 2015; Anselma and Mazzei, 2015).

3.1 La base di dati: Gedeone

Nel considerare il dominio di interesse è necessario pensare ai dati stessi su cui andiamo a lavorare. *Gedeone*[1] è un portale di Coop Italia dedicato a promuovere un sano stile di vita e dieta proponendo consigli, articoli e soprattutto ricette. Al fine del progetto MADiMAN, le informazioni presenti su questo sito sono state manipolate e trasposte in una base di dati con DBMS PostgreSQL, rendendo disponibili un gran quantitativo di ricette (circa 500), passi preparativi, ingredienti, valori nutrizionali e vari metadati.

Un'ulteriore componente dello schema analizzato include dati e tabelle su utenti e la loro dieta, lo storico dei pasti e delle pesate.

Per gli obiettivi di questo lavoro è stato deciso di considerare la parte del database relativa alle ricette trascurando la parte relativa alla pianificazione dei pasti considerando il punto di vista di un utente non esperto del funzionamento della memorizzazione dei dati, ancor meno di come questi siano strutturati internamente.

4 Architettura del prototipo LN→SQL

Primariamente è stato costruito un corpus di 12 domande in Linguaggio Naturale[2] (Tabella 1). Gli scopi di questo piccolo corpus sono principalmente due. Primo, nel progettare un sistema che si interfacci con una persona tramite un unico passo in un caso d'uso ipotetico (inserimento testuale), è stato necessario pensare a delle domande plausibili che un utente medio potrebbe porre tramite applicativo. È stato ipotizzato a questo fine che le informazioni più desiderabili derivino dalla possibilità di consultare il ricettario, ponendo domande implicitamente relative alla propria dieta personale (per esempio, sapere l'apporto calorico di un certo piatto) così come richiedendo istruzioni ed ingredienti. Secondo, per permettere il testing di un prototipo, ognuna di queste domande è stata scritta affinché verificasse l'abilità del sistema nel riconoscere e derivare all'interno della struttura sintattica particolari costrutti, strutture ed operazioni appartenenti alla query finale. Prendendo spunto da (Affolter et al., 2019), ogni domanda è stata categorizzata attraverso varie label, principalmente corrispondenti agli operatori SQL *Join*, *Filtraggio* su attributo (nella clausola WHERE), *Negazione*, *Ordinamento* esplicito, *Raggruppamento*, *Aggregazione*, *Sottoquery*.

#	Frase
1	Quali sono le ricette senza fritture?
2	Quali sono le ricette con meno di 500 kcal?
3	Quali sono i piatti con più di 40g di proteine?
4	Quali sono i piatti con meno carboidrati?
5	Quali sono le preparazioni delle ricette primaverili?
6	Quanta acqua ho consumato il giorno @data?
7	Quante calorie ho assunto ogni giorno dal @date1 al @date2?
8	Quali sono le ricette di pesce in ordine di difficoltà?
9	Quali sono le ricette facili col minor tempo di preparazione?
10	Quali sono le ricette con cottura in forno o a vapore?
11	Quali sono gli ingredienti per @people persone della ricetta @recipe?
12	Quali sono i piatti con pochi carboidrati?

Tabella 1: Le 12 domande nel dominio delle ricette di Gedeone ideate per la sperimentazione. Il simbolo @ indica delle variabili nella domanda.

A ogni domanda definita nel corpus è stata associata una corrispondente query SQL, la cui correttezza è stata verificata sul database. Mentre si è cercato il più possibile di semplificare il formato della traduzione finale di ogni domanda in input, la necessità a fini sperimentali di definire domande sempre più complesse ha portato ad ottenere alcune query con strutture articolate, il che ha talvolta reso arduo il processo di traduzione.

Il processo di traduzione di ogni domanda del corpus è sequenziale e si divide in una fase preliminare di preanalisi, e in due fasi principali concatenate fra di loro.

Nella preanalisi la domanda in italiano è suddivisa in token. Oltre a ciò sono riconosciuti ed estratti possibili valori di attributi relativi alla base di dati considerata ed è attuata dove necessario una sostituzione con sinonimi. Par-

[1] http://www.gedeone-e-coop.it

[2] Il corpus completo è disponibile online: http://www.di.unito.it/~mazzei/papers/clic2021/QnAcorpus.pdf

```
PP[PT=?pt, RL=?rl, LF=?fl, ORD=?ord]
  ->
P[PT=?pt] NP[RL=?rl, LF=?fl, ORD=?ord]

NN[ORD=<\x.asc(x, id)>] -> 'ordine'
```

Figura 1: Un frammento della FCFG usata per rappresentare la semantica di una domanda in Calcolo Relazionale.

tendo dalla terza frase della Tabella 1, ovvero *Quali sono i piatti con più di 40g di proteine?*, la preanalisi, dopo aver eseguito tokenizazione e sostituzione dei sinonimi, restituisce: `[quali, sono, i, ricetta, con, più, di, 40, grammi, di, proteineg]`.

La prima fase prende in input i token ottenuti in preanalisi e restituisce un albero sintattico annotato, ottenuto tramite il parsing dei token su una FCFG. È ottenuta, come rappresentazione intermedia, un'espressione nel Calcolo Relazionale su Tuple.

Infine, nella seconda fase, le informazioni presenti nell'albero sintattico sono estratte tramite diverse visite, durante ognuna delle quali è costruita una clausola della corrispondente query in SQL. Nel seguito dettagliamo il funzionamento delle due fasi principali e poi analizziamo le prestazioni del sistema.

4.1 Prima fase: dal Linguaggio Naturale al Calcolo Relazionale

In questo lavoro abbiamo usato delle grammatiche libere da contesto con feature[3] (FCFG) utilizzando la libreria Python NLTK (Bird, 2006). Similmente a (Warren and Pereira, 1982; Song et al., 2015), abbiamo inizialmente usato una rappresentazione intermedia di tipo logico per rappresentare il significato della domanda. L'obbiettivo era appunto quello di realizzarne una traduzione il più possibile fedele a una formula della FoL, la quale esprimesse precisamente il significato della query finale nel dominio (tabelle coinvolte, espressioni per la clausola WHERE, valori di attributi, ecc.).

Dopo le prime sperimentazioni si è riscontrato che questa forma intermedia, per quanto intuitiva e formale, non permetteva una facile traduzione finale nella seconda fase. Inoltre, fare affidamento esclusivamente su una singola formula composta limitava i potenziali vantaggi dati dall'uso delle FCFG. In questo cambio di direzione è stato riscontrato che una rappresentazione intermedia più vicina all'obbiettivo della traduzione in SQL poteva essere data da una query in calcolo relazionale. Il *Calcolo Relazionale su Tuple con dichiarazione di Range* è un linguaggio formale che permette di esprimere in modo dichiarativo una interrogazione su una base di dati relazionale. In questo formalismo una query comprende tre componenti: Target List (TL), Range List (RL) e Logic Formula (LF) (Codd, 1972) (cf. Fig. 1). Per ognuna di queste riportiamo una breve descrizione, seguita dal metodo scelto per poterla ottenere e rappresentare a partire dalla domanda in italiano e tramite la grammatica basata su feature.

Target List: specifica quali attributi compaiono nel risultato. Poiché il riconoscimento degli attributi necessari dipende sia da quali vengano esplicitati nella domanda, sia dalle tabelle coinvolte, questa componente è quasi completamente determinata nella seconda fase (a posteriori della parsing sintattico). Tramite le regole lessicali della grammatica sono riconosciuti gli eventuali attributi di tabelle, i cui simboli terminali sono annotati dalla loro rappresentazione nella feature LF, specificandone il tipo tramite la feature booleana +ATTR. Gli elementi restanti della Target List saranno in seguito derivati tramite una mappatura tabella-attributi.

Range List: specifica le variabili libere nella Formula Logica, cioè le tabelle su cui variano le variabili della LF per generare il risultato. Similarmente agli attributi, anche le tabelle possono essere riconosciute esplicitamente (ed annotate con una feature +TABLE), oppure derivate da attributi riscontrati, ma non appartenenti a nessuna tabella trovata fino a quel momento. In entrambi i casi la loro semantica è raccolta nella feature RL di una produzione terminale e, come per gli attributi, i diversi elementi saranno raccolti nella fase successiva attraversando l'albero sintattico annotato.

Logic Formula: specifica una formula che il risultato deve soddisfare. A differenza delle precedenti, la LF (feature LF) è composta ricorsivamente e i suoi predicati possono essere introdotti non solo nelle regole lessicali, ma anche nelle produzioni non terminali tramite il riconoscimento di particolari strutture sintattiche. La composizione della formula a partire da feature di nodi diversi è effettuata tramite le astrazioni del *lambda-calcolo*:

[3]La grammatica realizzata è disponibile online: `http://www.di.unito.it/~mazzei/papers/clic2021/calcolo_relazionale.fcfg`

definendo predicati parziali in un nodo è possibile applicarli a formule ed argomenti provenienti dagli altri nodi della stessa produzione in una semplificazione chiamata *Beta-riduzione*. Questa avviene automaticamente durante il parsing sintattico tramite NLTK, ed il risultato, se presente, è visibile nella radice dell'albero sintattico.

A differenza di SQL, il Calcolo Relazionale, per propria natura, non esprime esplicitamente un ordinamento sull'insieme di tuple risultante (corrispondente al costrutto *ORDER BY* di SQL). Per compensare questo aspetto si è deciso di utilizzare una feature aggiuntiva ORD, definita in modo simile alla feature LF ma utilizzata esclusivamente per mantenere tramite predicati le informazioni di un eventuale ordinamento esplicito (riconosciuto a livello lessicale da espressioni come "in ordine di", "con meno/più X", ecc.) In generale il processo di traduzione della prima fase è completamente automatizzato: il parser utilizzato è generato tramite una funzione della libreria NLTK e necessita solamente della definizione di una grammatica. Il parsing dei token in input restituisce dunque un albero sintattico annotato con feature.

Partendo dalla preanalisi della frase *Quali sono i piatti con più di 40g di proteine?*, si ottiene l'analisi sintattica in Fig. 2(a), e la corrispondente rappresentazione in calcolo relazionale in Fig. 2(b).

4.2 Seconda fase: dal Calcolo Relazionale a SQL

La costruzione dell'interrogazione SQL utilizza l'albero sintattico e la corrispondente formula del calcolo relazionale ottenute entrambe nella fase precedente, insieme a ulteriori dati provenienti dal dominio di interesse, e che sono implicitamente codificate nel database di riferimento. Mentre con la prima fase si è potuto modellare la conoscenza derivata dalla domanda iniziale, molto di ciò che ci serve sapere non è direttamente derivabile da una domanda in italiano. Inoltre, poiché l'obbiettivo di questo progetto rimane la creazione di un sistema che operi su un dominio definito, è necessario che le informazioni più importanti riguardanti questo siano definite ed accessibili nel sistema: queste informazioni sono fondamentali per stabilire i legami tra ciò che è stato chiesto e i nomi effettivi di tabelle e attributi.

Il processo di traduzione finale segue un algoritmo generale comune: per ogni clausola principale di SQL viene visitato l'albero sintattico e, grazie anche alle informazioni dal dominio, viene generata la clausola componendo ivalori nelle feature. In particolare:

SELECT: La lista di attributi da restituire corrisponde alla Target List del Calcolo Relazionale. È possibile riconoscere due tipi di attributi appartenenti a questa. (i) Ogni tabella introdotta, esplicitamente o meno (riconosciuta a partire dalla feature +TABLE oppure derivata da un suo attributo), comprende uno o più attributi di default, oltre a quelli di chiave primaria. Questo tipo di mappatura è un esempio di informazione pragmatica proveniente dal dominio e codificata nella tabella. (ii) Per quanto riguarda invece gli attributi non presenti in alcuna tabella, questi possono semplicemente essere riconosciuti da una regola lessicale (tramite la feature +ATTR) ed aggiunti in coda all'elenco.

FROM: Un'assunzione importante è stata fatta nella gestione dei join, ovvero che le tabelle avessero sempre delle chiavi esterne definite esplicitamente. Ciò semplifica notevolmente la traduzione, in quanto è sufficiente conoscere le tabelle coinvolte per poter generare l'intera clausola comprensiva delle condizioni di join. L'algoritmo di generazione riconosce i vincoli di chiave esterna tramite due cicli, che compongono la clausola risultante con gli attributi necessari. Da notare che sia i vincoli relazionali, sia gli alias delle tabelle sono definiti tramite dizionari.

WHERE: Questa clausola è l'unica costruita a partire dalle informazioni provenienti dalla LF. I predicati di questa sono indicati in un formalismo più simile a quello della FOL rispetto al Calcolo Relazionale, al fine di aderire alla sintassi delle formule di NLTK. Per ognuno dei predicati si effettua una traduzione corrispondente rappresentazione in SQL.

ORDER BY: Come la clausola WHERE, la costruzione di questa clausola è determinata dal valore della feature ORD nella radice dell'albero sintattico. Per i fini della nostra implementazione è stato deciso di adottare, nel caso in cui la feature ORD risulti vuota, un ordinamento di default sull'attributo di chiave primaria della tabella principale coinvolta.

Considerando ancora la terza frase della Tabella 1, il risultato della seconda fase di traduzione è in Fig. 2-(c).

```
(S[FL=<maggiore(ric(ricetta),proteineg,40)>, ORD=?ord]
  (VP[] (PI[NUM='pl', QRT='qual'] Quali) (IV[NUM='pl'] sono))
  (NP[FL=?fl]
    (DET[NUM='pl'] i)
    (NN[RL=<ric(ricetta)>, +TABLE] ricetta))
  (PP[FL=<maggiore(ric(ricetta),proteineg,40)>, ORD=?ord, PT='with', RL=?rl]
    (P[PT='with'] con)
    (NP[FL=<maggiore(ric(ricetta),proteineg,40)>]
      (ADV[FL=<\z y x.maggiore(x,y,z)>] più)
      (PP[FL=<40>, ORD=?ord, PT='of', RL=?rl]
        (P[PT='of'] di)
        (NP[FL=<40>] (CARD[FL=<40>] 40) (NN[] grammi)))
      (PP[FL=<proteineg>, PT='of', RL=<ric(ricetta)>]
        (P[PT='of'] di)
        (NN[+ATTR, FL=<proteineg>, RL=<ric(ricetta)>] proteineg)))))
```

(a)

```
{TL=ric.id, ric.descrizione, ric.nome , ric.proteineg | RL=ric(ricetta) |
LF=maggiore(ric(ricetta),proteineg,40)}
```

(b)

```
SELECT ric.id, ric.nome, ric.descrizione, ric.proteineg
FROM Ricetta ric WHERE ric.proteineg>40 ORDER BY ric.id ASC;
```

(c)

Figura 2: Il risultato della prima fase di analisi (a-b) e della seconda fase di analisi (c) sulla frase *Quali sono i piatti con più di 40g di proteine?*.

5 Valutazione preliminare

Nella sperimentazione preliminare è stato possibile implementare la traduzione solamente di 7 domande rispetto alle 12 inizialmente definite nel corpus[4]. Seguono i limiti incontrati per alcune delle interrogazioni non realizzate.

La domanda 6, *Quanta acqua ho consumato il giorno @data*, è stata tralasciata poiché richiederebbe una gestione delle informazioni riguardanti un utente che abbia effettuato l'accesso al sistema.

Non è stato possibile realizzare la domanda 10, *"Quali sono le ricette con cottura in forno o a vapore"* per la natura dell'operazione richiesta (un OR esclusivo), la quale richiedeva una sotto-query. Al momento non è implementato un meccanismo che rilevi la presenza di sotto-interrogazioni a partire dalla struttura della frase.

Nella domanda 11, *"Quali sono gli ingredienti per @people persone della ricetta @recipe?"*, la difficoltà incontrata è stata la gestione dei valori NULL ottenibili in certi attributi.

La domanda 12, *"Quali sono i piatti con pochi carboidrati?"*, è stata realizzata nella versione semplificata *"Quali sono i piatti con meno carboidrati?"*: la difficoltà nell'includere una sottoquery (con annessa Window Function di SQL) ha portato a interpretare "pochi carboidrati" non come valori appartenenti al primo quartile sulla distribuzione totale, ma come quantità al di sotto di una soglia di default.

[4]Un elenco completo dell'output è disponibile a `http://www.di.unito.it/~mazzei/papers/clic2021/OutputPrototipo.pdf`

6 Conclusioni

In questo lavoro sono stati presentati i primi risultati di un progetto ancora in corso per interpretare una domanda in linguaggio naturale come un'interrogazione SQL nel dominio della dieta mediterranea. Le due caratteristiche principali di questo progetto sono state l'uso del calcolo relazionale su tuple per rappresentare il significato della frase (Sezione 4.1) e l'uso delle specificità del dominio per trasformare poi tale rappresentazione in SQL (Sezione 4.2).

In futuro intendiamo completare le interrogazioni contenute nel corpus proposto per poi integrare il prototipo in un agente conversazionale sul dominio della gestione di una dieta salutare. Considerando la presenza di indeterminatezza nel linguaggio naturale, potrebbe essere interessante dare supporto per estensioni del modello relazionale che trattano indeterminatezza (Anselma et al., 2016).

References

Katrin Affolter, Kurt Stockinger, and Abraham Bernstein. 2019. A comparative survey of recent natural language interfaces for databases. *The VLDB Journal*, 28(5):793–819.

Sihem Amer-Yahia, Georgia Koutrika, Frederic Bastian, Theofilos Belmpas, Martin Braschler, Ursin Brunner, Diego Calvanese, Maximilian Fabricius, Orest Gkini, Catherine Kosten, Davide Lanti, Antonis Litke, Hendrik Lücke-Tieke, Francesco Alessandro Massucci, Tarcisio Mendes de Farias, Alessandro Mosca, Francesco Multari, Nikolaos Papadakis, Dimitris Papadopoulos, Yogendra Patil, Aurélien Personnaz, Guillem Rull, Ana Claudia Sima, Ellery Smith, Dimitrios Skoutas, Srividya Subramanian, Guohui Xiao, and Kurt Stockinger. 2021. INODE: building an end-to-end data exploration system in practice [extended vision]. *CoRR*, abs/2104.04194.

Luca Anselma and Alessandro Mazzei. 2015. Towards diet management with automatic reasoning and persuasive natural language generation. In *Portuguese Conference on Artificial Intelligence*, pages 79–90. Springer.

Luca Anselma and Alessandro Mazzei. 2020. Building a persuasive virtual dietitian. *Informatics*, 7(3):27.

Luca Anselma, Luca Piovesan, and Paolo Terenziani. 2016. A 1nf temporal relational model and algebra coping with valid-time temporal indeterminacy. *Journal of Intelligent Information Systems*, 47(3):345–374.

Luca Anselma, Alessandro Mazzei, and Andrea Pirone. 2018. Automatic reasoning evaluation in diet management based on an italian cookbook. In *Proceedings of the Joint Workshop on Multimedia for Cooking and Eating Activities and Multimedia Assisted Dietary Management*, pages 59–62.

Simone Balloccu, Ehud Reiter, Matteo G. Collu, Federico Sanna, Manuela Sanguinetti, and Maurizio Atzori. 2021. Unaddressed challenges in persuasive dieting chatbots. In Judith Masthoff, Eelco Herder, Nava Tintarev, and Marko Tkalcic, editors, *Adjunct Publication of the 29th ACM Conference on User Modeling, Adaptation and Personalization, UMAP 2021, Utrecht, The Netherlands, June 21-25, 2021*, pages 392–395. ACM.

Steven Bird. 2006. NLTK: the natural language toolkit. In Nicoletta Calzolari, Claire Cardie, and Pierre Isabelle, editors, *ACL 2006, 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July 2006*. The Association for Computer Linguistics.

E. F. Codd. 1972. Relational completeness of data base sublanguages. *Research Report / RJ / IBM / San Jose, California*, RJ987.

Dan Jurafsky. 2014. *The language of food: A linguist reads the menu*. WW Norton & Company.

Esther Kaufmann, Abraham Bernstein, and Renato Zumstein. 2006. Querix: A natural language interface to query ontologies based on clarification dialogs. In *5th international semantic web conference (ISWC 2006)*, pages 980–981.

Alessandro Mazzei, Luca Anselma, Franco De Michieli, Andrea Bolioli, Matteo Casu, Jelle Gerbrandy, and Ivan Lunardi. 2015. Mobile computing and artificial intelligence for diet management. In *International Conference on Image Analysis and Processing*, pages 342–349. Springer.

Dezhao Song, Frank Schilder, Charese Smiley, Chris Brew, Tom Zielund, Hiroko Bretz, Robert Martin, Chris Dale, John Duprey, Tim Miller, et al. 2015. Tr discover: A natural language interface for querying and analyzing interlinked datasets. In *International Semantic Web Conference*, pages 21–37. Springer.

David H.D. Warren and Fernando C.N. Pereira. 1982. An efficient easily adaptable system for interpreting natural language queries. *American Journal of Computational Linguistics*, 8(3-4):110–122.

Are Crescia and Piadina the Same? Towards Identifying Synonymy or Non-Synonymy between Italian Words to Enable Crowdsourcing from Language Learners

Lavinia Aparaschivei[1,2], **Lionel Nicolas**[2], **Alberto Barrón-Cedeño**[1]
[1] DIT–Università di Bologna, Forlì, Italy
[2] Institute for Applied Linguistics, Eurac Research, Bolzano, Italy
{lavinianicoleta.aparaschivei, lionel.nicolas}@eurac.edu,
a.barron@unibo.it

Abstract

We introduce a method to generate candidate pairs of related Italian words sharing (or not) synonymous relations from the ConceptNet knowledgebase. The pairs are intended to generate questions for a vocabulary trainer which combines exercises to enhance vocabulary skills with the implicit crowdsourcing of linguistic knowledge about the semantic relations between words. Our method relies on the idea that pairs of synonyms in a language tend to translate to pairs of synonyms in other languages. We generated $85k$ candidate pairs of Italian synonyms that can be used to produce questions for both teaching ($3.8k$ pairs) and crowdsourcing purposes ($80k$ pairs). Follow-up efforts are however needed in order to generate a complementary set of questions.

1 Introduction

Our efforts target the automatic generation of semantically-related candidate pairs of Italian words with a focus on synonymy. We address a cold start issue for a vocabulary trainer combining exercises to enhance vocabulary skills with the implicit crowdsourcing of linguistic knowledge about the semantic relations between words.

While targeting a specific use case, our method contributes to a larger effort aimed at narrowing gaps on two fronts. On the NLP front, over the past few decades varied efforts have targeted the efficient creation, extension, and maintenance of resources, including crowdsourcing through platforms such as Amazon Mechanical Turk (Ganbold et al., 2018; Potthast et al., 2018). Still, the subject remains an open issue. On the computer-assisted language learning (CALL) front, the automatic generation of exercise content from NLP resources is almost non-existent, despite the fact that some of these datasets encode the knowledge that learners are often tested on (e.g., lexical knowledge). This absence is probably due to differences in expectations with respect to linguistic accuracy: learning materials are usually close to perfect, whereas NLP resources rarely are. Generating content from imperfect datasets poses a challenge in terms of its suitability for learning.

We contribute to narrowing these gaps by producing data to tackle a cold start issue for a vocabulary trainer designed to both teach language and crowdsource linguistic knowledge from learners. We generate a collection of candidate pairs of Italian words tied to confidence scores, allowing to decide which pairs should be used for learning or for crowdsourcing purposes. Our method projects synonymy information in ConceptNet (Speer et al., 2017) from non-Italian onto Italian words. The obtained results show that we adequately tackle part of the cold start issue, while follow-up efforts are needed to address the remaining part.

The rest of the paper is organised as follows: Section 2 discusses the specific purpose of our method. Section 3 summarises related work. Section 4 and Section 5 describe how the candidate pairs are generated and scored. Finally, Section 6 discusses how suitable the pairs are for our specific use case and Section 7 provides closing remarks.

2 Background

Ours and previous related work (Lyding et al., 2019; Rodosthenous et al., 2019; Rodosthenous et al., 2020; Nicolas et al., 2021) all contribute to a wider effort to research an implicit crowdsourcing paradigm built upon the idea that the curation of NLP resources and language learning are sibling endeavors (Nicolas et al., 2020). On the one hand, (NLP) researchers try to create models to "teach" a computer to process and/or produce language utterances. On the other hand, learners create a model, in the form of personal knowledge, to process and/or produce language utterances too. Allowing learners to express their knowledge can contribute to enhance an NLP resource, under specific conditions. This paradigm substitutes the expert manpower typically required to curate NLP resources with a non-expert crowd of learners.

Expert manpower can be substituted by non-expert crowds, as exemplified by the numerous efforts to use Amazon Mechanical Turk (AMT) to build NLP datasets; e.g., Ganbold et al. (2018) or Potthast et al. (2018). The synergy exploited by this paradigm between NLP and CALL is particularly interesting. Indeed, NLP can be used to enhance CALL methods and, as such, it can grant an intrinsic added value for the target crowd that is not limited by any type of resource, unlike other crowdsourcing approaches relying on extrinsic added values (e.g. monetary incentives in AMT). In addition, from the NLP perspective, the crowd of learners that could potentially be reached is immense. Accordingly, an unprecedented amount of data could theoretically be crowdsourced by exploiting such a synergy.

Nicolas et al. (2021) showed that linguistic knowledge about an entry of an NLP dataset can be obtained at expert quality level, provided that the same judgement is asked to a sufficient number of learners. This is mostly true when simple Boolean questions are used. Even linguistic judgements of inferior reliability (e.g. 70%) contribute to approaching statistical certainty about the right answer to Boolean questions.[1] This approach favors quantity over quality to meet its goals and can be used to produce new entries or to validate the existing ones in an NLP dataset.

V-trel is a vocabulary trainer that implements this paradigm to teach and crowdsource knowledge on semantic relations between words (Lyding et al., 2019; Rodosthenous et al., 2019; Nicolas et al., 2021). V-trel includes two types of questions. Open questions ask for words sharing a specific relation with a given one (e.g., "give me a synonym of x"). Closed questions show a pair of words and ask the Boolean question of whether they share a specific relation (e.g., "are x and y synonyms?"). From the crowdsourcing perspective, open questions are mostly intended to crowdsource additional knowledge (i.e. to crowdsource new candidate entries), whereas closed questions are designed to crowdsource judgements on the knowledge suggested in the open questions or already encoded in ConceptNet (i.e. to validate existing entries or new candidate entries).

As empirically observed, closed questions should elicit positive and negative answers from learners (Rodosthenous et al., 2020). Otherwise, when learners understand that the trainer tends to continuously expect the same answer (e.g., "yes"), they tend to give the same default answer mechanically, without producing meaningful judgments.

We aim at generating closed questions expecting both types of answers. We need to identify pairs of synonyms to produce questions eliciting a positive answer and word pairs sharing a semantic relation other than synonymy (e.g. antonymy) to elicit negative answers. We refer to them as *non-synonyms* in the rest of the paper. To elicit positive answers, we use synonyms, such as "house" and "home". To elicit negative answers, we need non-synonyms such as "good" and "bad". It is worth noting that we do not consider as non-synonyms pairs of unrelated words such as "house" and "dog"' because they do not share any kind of semantic relation. Questions eliciting negative answers generated from them would be of poor quality and would not pose any challenge to learners.

Since v-trel favours teaching over crowdsourcing to maintain its pool of users, closed questions designed to crowdsource knowledge from learners should be served on a low frequency. This implies the need to decide which questions can be used for teaching and which for crowdsourcing purposes. Hence, the need of a confidence score to divide the questions into the two sets.

Our method aims at replacing the closed questions generated from ConceptNet, whose expected answers have quality issues since ConceptNet is, as most NLP resources, an imperfect dataset and for which we cannot tell apart the questions that can be used for teaching and for crowdsourcing purposes. Even though the aggregation of answers crowdsourced from learners would solve

[1] While the results obtained tend to confirm the viability of the approach, many aspects remain unexplored; e.g., the difficulty of a question in the aggregation process.

these problems, we have a cold start issue similar to the chicken and egg paradox: the issues cannot be solved without offering the tool but the tool cannot be offered without solving the issues first.

3 Related Work

With respect to the automatic generation of language learning exercises, only little automatic generation is performed directly from NLP resources so far. Most efforts focus on exercises known as a "cloze" (deletion) test, where learners have to fill word gaps in a text (Hill and Simha, 2016; Katinskaia et al., 2018; Lee et al., 2019). The literature from the last four editions of the top-two venues concerned with using NLP for CALL[2] confirms that current efforts, aside from ours, are mostly dedicated to the generation of cloze exercises (Santhi Ponnusamy and Meurers, 2021), the modelling of the learner knowledge (Araneta et al., 2020), or the detection and/or correction of mistakes in written text (Üksik et al., 2021). Some preliminary efforts exist on the automatic generation of exercises from Finnish and Hungarian NLP resources.[3] Despite the relatively narrow nature of the exercises we aim at generating[4], our work represents one of the few efforts targeting the automatic generation of language learning exercises from NLP resources.

Since our method generates pairs of synonyms from an existing knowledge base, it shares common ground with approaches to build or extend similar datasets. In that respect, the state of the art is mostly concerned with the creation and curation of WordNets for which various semi- and fully-automatic techniques have been developed, especially for languages other than English. Following Vossen (1996), these methods can be categorised as using either a *merge* or an *expansion* approach or both. The merge approach employs monolingual resources to create a standalone WordNet and was adopted for EuroWordNet (Vossen, 1998), the Polish WordNet (Derwojedowa et al., 2008), the Norwegian WordNet (Fjeld and Nygaard, 2009) and the Danish WordNet (Pedersen et al., 2009). The expansion approach uses a source WordNet and translates its synsets into the target language. It was used to build MultiWordNet (Pianta et al., 2002), the Finnish WordNet (Linden and Carlson, 2010), the French WordNet WOLF (Sagot and Fišer, 2008), and to enhance a Persian WordNet (Mousavi and Faili, 2017; Mousavi and Faili, 2021).

Our method employs an expansion approach: it projects knowledge from other languages onto Italian, but it differs in three aspects. First, it relies on a different type of dataset: ConceptNet.[5] Second, the output is not a final product, but a "raw" dataset to be polished by crowdsourcing. Third, it aims at identifying both synonyms and non-synonyms, whereas the aforementioned methods are mostly concerned with synonyms only.

4 Generating Candidate Pairs

Our hypothesis is that if two non-Italian words are marked as synonyms in ConceptNet and such words are translations of a pair of Italian words, then the Italian words are synonyms with a high likelihood. For instance, the pair {*house*, *home*} in English with respect to {*casa*, *abitazione*} in Italian. The greater the number of such pairs of non-Italian words are identified (e.g., {*maison*, *logement*} in French, {*casa*, *vivienda*} in Spanish), the more likely the Italian words are to be synonymous. Hereafter, we refer to the number of pairs of non-Italian words projected onto an Italian pair as *Nb-projected-syn-pairs*.

At the same time, we assumed that the incorrect candidate pairs of synonyms generated would mostly constitute a valid set of candidate pairs of non-synonyms. As such, most candidate pairs would be used to tackle our specific use case.

We used this logic for all languages available in ConceptNet. In order to seamlessly add the data already available on Italian synonyms, we considered the Italian part of ConceptNet as describing just another non-Italian language. Hence, we considered all Italian words as translations of themselves in this "extra" language.

We extracted $84,602$ candidate pairs of Italian synonyms and randomly sampled and evaluated a subset of $1,120$ pairs to build a gold standard.

[2] The BEA Workshop `https://aclanthology.org/venues/bea/`, and the NLP4CALL Workshop `https://aclanthology.org/venues/nlp4call/`

[3] See the following PhD project: `https://spraakbanken.gu.se/cms/sites/default/files/2021/nlp4call2021_researchnotes1_talk1.pdf`

[4] It would certainly be interesting to extend such exercises with a sentence context.

[5] ConceptNet is a multilingual knowledge base that represents commonly-used words and phrases as well as the relationships between them. It currently holds more than 34 million assertions about words: $\texttt{term}_a$ `<relation>` $\texttt{term}_b$. ConceptNet can be accessed via an API, making it easy to integrate into applications.

The annotation procedure started by reflecting the information in well-known online Italian dictionaries: Treccani, De Mauro, Gabrielli, Sabatini-Coletti, Rizzoli, and Virgilio.[6] When a candidate pair was not found in these dictionaries, an annotator studied the definitions of the two words and searched for a third word referenced as a synonym of both words in the pair. We only kept instances where the annotator showed a high confidence. In total, 515 were labeled as correct pairs and 485 as incorrect. We discarded 120 pairs. From the $1,000$ annotated instances, 403 directly reflect the information of reference dictionaries, whereas 597 reflect the stand of the annotator.

By extrapolating the ratio observed in the gold standard, we estimate that 51.2% of the candidate pairs ($\sim 43.4k$ pairs) are indeed synonyms. In comparison, 19,906 Italian word pairs are marked as synonyms in ConceptNet. We randomly sampled and annotated 200 of them with the procedure used to build the gold standard. Our estimation that 84% of them ($\sim 16.7k$ pairs) are valid. Our set of candidate pairs of Italian synonyms is thus larger, but has lower quality. Using these pairs directly to generate closed questions eliciting a positive answer would thus defeat our goal of improving the quality of the closed questions.

5 Computing Confidence Scores

We aim at discriminating between instances intended to generate questions eliciting positive and negative answers, while discriminating questions used for teaching or crowdsourcing purposes. We relied on a binary classifier to flag candidate pairs as correct and incorrect instances of synonyms. The predictions are used to decide on the kind of answers to elicit —candidate pairs predicted as correct are used to generate questions eliciting a positive answer and vice-versa. The associated confidence scores are used to discriminate between questions used for teaching and for crowdsourcing. We used the aforementioned gold standard to train the classifier.

The features are the following. (1) The aforementioned *Nb-projected-syn-pairs* for each pair. (2) To distinguish the languages from which the projection of knowledge happened, we computed per language the size of each subset of pairs of non-Italian words projected onto the candidate pair (which, together, sum up to *Nb-projected-syn-pairs*). (3) To express "relatedness", we computed the size of the set of non-Italian pairs of words both marked as sharing a semantic relation (i.e. not only synonymy) and as translations of the candidate pair. We refer hereafter to this number as *Nb-projected-all-pairs*. We also computed a ratio obtained by dividing *Nb-projected-syn-pairs* by *Nb-projected-all-pairs*. (4) To indicate if a candidate pair might be better suited to another semantic relation, we computed per semantic relation the size of the subsets of non-Italian pairs of words both marked as sharing a semantic relation other than synonymy and as translations of the candidate pair, as well as a ratio value by dividing these sizes by *Nb-projected-syn-pairs* and a difference by subtracting *Nb-projected-syn-pairs* to them. (5) A last set of features represents the most found relation, besides synonymy, in these non-Italian pairs of words (i.e. the top "competitor") by providing its type and duplicating the corresponding size of the subset of non-Italian pairs of words, ratio, and difference.

model	precision	recall	F_1
Random forest	**72.2**	62.4	67.0
Logistic regression	61.1	79.8	**69.2**
Random Tree	71.9	60.8	65.9
Baseline	51.2	**100.0**	67.7

Table 1: Leave-one-out cross-validation performance of three classifiers when identifying correct synonym pairs against an all-correct baseline.

Since our gold standard is small, we ran a leave-one-out cross validation process to assess the quality of the predictions for a number of classifiers with default settings.[7] Table 1 shows the performance obtained by three of them plus a baseline that labels all pairs as correct. Even if the logistic regressor obtains the highest F_1, we adopt the model with the highest precision: the random forest. The reason is that we have observed empirically that precision is the most adequate indicator of how much the confidence scores would corre-

[6] https://www.treccani.it; https://dizionario.internazionale.it; https://www.grandidizionari.it/Dizionario_Italiano/; https://dizionari.corriere.it/dizionario_italiano/; https://dizionari.corriere.it/dizionario_sinonimi_contrari/; https://sapere.virgilio.it.

[7] We used Weka 3.8.8; https://www.cs.waikato.ac.nz/ml/weka/.

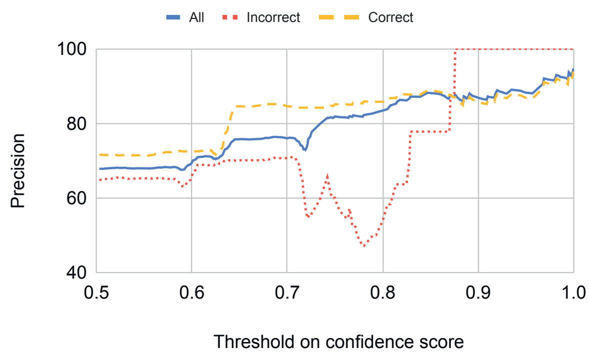


Figure 1: Precision against confidence score threshold (lower-score predictions neglected).

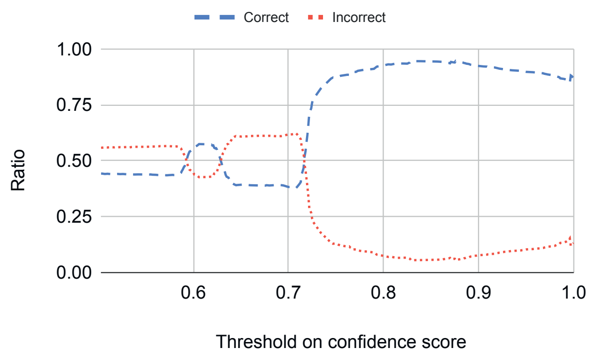


Figure 2: Ratio of label predicted according to thresholds on confidence scores (lower-score predictions neglected).

late with the quality of the predicted labels.[8]

6 Categorising Candidate Pairs

Once the binary classification was completed, we had to distinguish which pairs could be used to generate teaching and which to generate crowdsourcing questions. For that, we studied the correlation between confidence scores and quality of prediction.

Figure 1 shows the precision obtained when thresholding at different confidence score values. The "all" curve shows a clear correlation between the quality of the label predicted and the confidence scores, which was the main result we were aiming for. However, the performance differs noticeably with respect to the label predicted: the curve associated with pairs predicted as "correct" grows as expected, whereas the one for pairs predicted as "incorrect" does not. The reason can be observed through the ratio of labels predicted according to confidence scores.

As Figure 2 shows, label "incorrect" was rarely predicted with high confidence scores. This is because our method is inherently oriented towards identifying pairs of synonyms. Accordingly, the pairs outputted that are not synonyms are also not, as we hoped for, pairs of non-synonyms. They are mostly random noise induced by homonyms in other languages. For example, the candidate {`fuoco`, `licenziare`} was generated because the English words {`fire`, `dismiss`} are synonyms. `Fire` has several homonyms with different senses, one of which translates to `fuoco` in Italian. Our set of candidate pairs thus contains only a few non-synonym pairs that the binary classifier struggles to spot. Therefore, our method cannot be used at present to generate closed questions eliciting negative answers.

This is not the case for pairs predicted as correct. For example, by using a minimum threshold of 0.996 on the confidence scores, we can select 3,829 pairs for which the predicted "correct" label is 94.44% reliable. This represents a set of pairs of reasonable size and better quality than the ones encoded in ConceptNet, which allow us to address part of the cold-start issue.

7 Conclusions and Ongoing Work

We presented a method to generate candidate pairs of Italian words that are synonyms or non-synonyms of one another from ConceptNet. These pairs will be used to generate questions used by a vocabulary trainer designed to combine the crowdsourcing of NLP datasets with language learning. While overtime all questions will be used for both teaching and crowdsourcing purposes, part of the pairs generated will at first be used to teach learners while the other part will at first be used to crowdsource knowledge in order to enhance ConceptNet. The obtained pairs, known to be correct synonyms in advance, can be served to the learners to improve their vocabulary skills. Another subset, whose correctness is still to be confirmed, can be served to the learners for validation and to decide whether the synonym connection between them should be added to ConceptNet or not.

Our results show that we can produce adequate data to generate part of the questions, while we are still unable to produce the data required to

[8]Future efforts will explore more direct and quantifiable means of formally informing this selection; cf. Section 7.

generate the complementary set of questions. In order to tackle the latter, we are devising a similar approach to identify candidate pairs of non-synonyms. We are adapting our overall procedure for the pairs of Italian words marked as translations of non-Italian words sharing any semantic relations (e.g. antonyms or hyponyms) instead of only considering the ones marked as translations of non-Italian words sharing a synonymy relation.

We are also interested in exploring possibilities to perform a more informed selection of the binary classification algorithm and will explore metrics to quantify the correlation between confidence scores and the quality of the predicted labels (e.g. Pearson, Kendall). In the future, we aim at running a crowdsourcing experiment with students of Italian as a second language with the produced data.

References

Marianne Grace Araneta, Gülşen Eryiğit, Alexander König, Ji-Ung Lee, Ana Luís, Verena Lyding, Lionel Nicolas, Christos Rodosthenous, and Federico Sangati. 2020. Substituto – a synchronous educational language game for simultaneous teaching and crowdsourcing. In *Proceedings of the 9th Workshop on NLP for Computer Assisted Language Learning*, pages 1–9, Gothenburg, Sweden, November. LiU Electronic Press.

Magdalena Derwojedowa, Maciej Piasecki, Stanisªaw Szpakowicz, Magdalena Zawisławska, and Bartosz Broda. 2008. Words, concepts and relations in the construction of polish wordnet. *Proceedings of GWC 2008*, pages 162–177.

Ruth Vatvedt Fjeld and Lars Nygaard. 2009. Nornet-a monolingual wordnet of modern norwegian. In *NODALIDA 2009 workshop: WordNets and other Lexical Semantic Resources-between Lexical Semantics, Lexicography, Terminology and Formal Ontologies*, volume 7, pages 13–16.

Amarsanaa Ganbold, Altangerel Chagnaa, and Gábor Bella. 2018. Using crowd agreement for wordnet localization. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Jennifer Hill and Rahul Simha. 2016. Automatic generation of context-based fill-in-the-blank exercises using co-occurrence likelihoods and google n-grams. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 23–30.

Anisia Katinskaia, Javad Nouri, Roman Yangarber, et al. 2018. Revita: a language-learning platform at the intersection of its and call. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Ji-Ung Lee, Erik Schwan, and Christian M Meyer. 2019. Manipulating the difficulty of c-tests. *arXiv preprint arXiv:1906.06905*.

K Linden and L Carlson. 2010. Construction of a finnwordnet. *Nordic Journal of Lexicography*, 17:119–140.

Verena Lyding, Christos Rodosthenous, Federico Sangati, Umair ul Hassan, Lionel Nicolas, Alexander König, Jolita Horbacauskiene, and Anisia Katinskaia. 2019. v-trel: Vocabulary trainer for tracing word relations - an implicit crowdsourcing approach. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 674–683, Varna, Bulgaria, September. INCOMA Ltd.

Zahra Mousavi and Heshaam Faili. 2017. Persian wordnet construction using supervised learning. *arXiv preprint arXiv:1704.03223*.

Zahra Mousavi and Heshaam Faili. 2021. Developing the persian wordnet of verbs using supervised learning. *Transactions on Asian and Low-Resource Language Information Processing*, 20(4):1–18.

Lionel Nicolas, Verena Lyding, Claudia Borg, Corina Forăscu, Karën Fort, Katerina Zdravkova, Iztok Kosem, Jaka Čibej, Špela Arhar Holdt, Alice Millour, et al. 2020. Creating expert knowledge by relying on language learners: a generic approach for mass-producing language resources by combining implicit crowdsourcing and language learning. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 268–278.

Lionel Nicolas, Lavinia Nicoleta Aparaschivei, Verena Lyding, Christos Rodosthenous, Federico Sangati, Alexander König, and Corina Forascu. 2021. An experiment on implicitly crowdsourcing expert knowledge about Romanian synonyms from language learners. In *Proceedings of the 10th Workshop on NLP for Computer Assisted Language Learning*, pages 1–14, Online, May. LiU Electronic Press.

Bolette Sandford Pedersen, Sanni Nimb, Jørg Asmussen, Nicolai Hartvig Sørensen, Lars Trap-Jensen, and Henrik Lorentzen. 2009. Dannet: the challenge of compiling a wordnet for danish by reusing a monolingual dictionary. *Language resources and evaluation*, 43(3):269–299.

Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. 2002. Multiwordnet: developing an aligned multilingual database. In *First international conference on global WordNet*, pages 293–302.

Martin Potthast, Tim Gollub, Kristof Komlossy, Sebastian Schuster, Matti Wiegmann, Erika Patricia Garces Fernandez, Matthias Hagen, and Benno Stein. 2018. Crowdsourcing a large corpus of clickbait on twitter. In *Proceedings of the 27th international conference on computational linguistics*, pages 1498–1507.

Christos T. Rodosthenous, Verena Lyding, Alexander Koenig, Jolita Horbacauskiene, Anisia Katinskaia, Umair ul Hassan, Nicos Isaak, Federico Sangati, and Lionel Nicolas. 2019. Designing a prototype architecture for crowdsourcing language resources. In *LDK*.

Christos Rodosthenous, Verena Lyding, Federico Sangati, Alexander König, Umair ul Hassan, Lionel Nicolas, Jolita Horbacauskiene, Anisia Katinskaia, and Lavinia Aparaschivei. 2020. Using crowdsourced exercises for vocabulary training to expand conceptnet. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 307–316.

Benoît Sagot and Darja Fišer. 2008. Building a free french wordnet from multilingual resources. In *OntoLex*.

Haemanth Santhi Ponnusamy and Detmar Meurers. 2021. Employing distributional semantics to organize task-focused vocabulary learning. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 26–36, Online, April. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*.

Tiiu Üksik, Jelena Kallas, Kristina Koppel, Katrin Tsepelina, and Raili Pool. 2021. Estonian as a second language teacher's tools. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 130–134, Online, April. Association for Computational Linguistics.

Piek Vossen. 1996. Right or wrong: Combining lexical resources in the eurowordnet project. In *Proceedings of the 7th EURALEX International Congress*, pages 715–728, aug.

Piek Vossen. 1998. Introduction to eurowordnet. In *EuroWordNet: A multilingual database with lexical semantic networks*, pages 1–17. Springer.

Moving from Human Ratings to Word Vectors to Classify People with Focal Dementias: Are We There Yet?

Chiara Barattieri di San Pietro[1,2], **Marco Marelli**[1], **Carlo Reverberi**[1]
1. Università degli Studi di Milano-Bicocca, Milano, Italy
2. Università degli Studi di Verona, Verona, Italy
chiara.barattieridisanpietro@unimib.it,
carlo.reverberi@unimib.it, marco.marelli@unimib.it

Abstract

Fine-grained variables based on semantic proximity of words can provide helpful diagnostic information when applied to the analysis of Verbal Fluency tasks. However, before leaving human-based ratings in favour of measures derived from distributional approaches, it is essential to assess the performance of the latter against that of the former. In this work, we analysed a Verbal Fluency task using measures of semantic proximity derived from Distributional Semantic Models of language, and we show how Machine Learning models based on them are less accurate in classifying patients with focal dementias than the same models built on human-based ratings. We discuss the possible interpretation of these results and the implications for the application of distributional semantics in clinical settings.

1 Introduction

A Verbal Fluency (VF) task (Lezak et al., 2004) is a test routinely used in the neuropsychological practice that requires participants to produce as many words as possible belonging to a given semantic category (e.g., "colours, "animals", etc.) within a time limit (typically 60 sec). It is commonly used to study lexical retrieval, and the subject's performance is standardly rated by the number of correct words produced for a given cue. However, to overcome the opacity of the overall score and help distinguish the different cognitive functions underpinning VF performance, additional measures of VF performance have been proposed. Among these, the number of consecutive words produced that share similar properties such as being a citrus fruit (this is called "semantic cluster" and its size is a clinically useful variable), and the total number of transitions between clusters (called "number of switches" – Troyer et al., 1997). Indeed, by characterising a semantic VF task (category "fruits") using the number of semantic categories produced, the average semantic proximity between words, the number of new words and out-of-category words, it has been possible to classify people with and without focal dementias, as well as across three different subtypes of dementias (Fronto-Temporal Dementia *versus* Primary Progressive Aphasia *versus* Semantic Dementia) with good accuracy (78% accuracy for patients vs healthy control classification, and 58.3% accuracy for classification across three pathological subcategories – Reverberi et al., 2014). One shortcoming of this model, however, is that those VP indexes are built upon human-based ratings of semantic proximity between pairs of words collected from a sample of healthy controls, making it hard to extend the same approach to words for which human judgments were not previously collected, i.e., other semantic categories.

Recent advances in Natural Language Processing techniques could help overcome this limitation. Distributional Semantic Models (DSMs) of language start from lexical co-occurrences extracted from large text corpora (Turney & Pantel, 2010), and applying different computational techniques, end up representing word meanings as numerical vectors in a multidimensional space. Here, terms that are semantically related are located close to each other. Such models can be used to simulate the structure of conceptual knowledge implied in the performance of semantic tasks such as a VF task. Indeed, DSMs have been successfully applied to different tasks of semantic relationships (Mandera et al., 2017), including the analysis of VF tasks to classify patients with Alzheimer's disease (Linz et al., 2017) and reaching remarkable accuracy (F1 = 0.77). However, despite the success, questions have been posed concerning what exactly distributional models can learn (Erk, 2016) and if such models are sufficiently rich in terms of encoded features (Lucy and Gauthier, 2017) to be applied to all sorts of semantic tasks/problems.

The present study aims to test if the analysis of a VF task based on DSM-derived measures would reproduce the results of an analysis based on human-derived measures. In particular, we decided to re-analyse the original data of a semantic VF task (category "fruit") that Reverberi et al. collected on a cohort of participants with focal dementias and healthy controls (CTR). Focal dementias are neurodegenerative diseases that cause deterioration of cognitive function, including language. The original cohort included people with Fronto-Temporal Dementia (FTD), Primary Progressive Aphasia (PPA), and Semantic Dementia (SD). Each diagnostic group presents peculiar linguistic symptomatology, making these syndromes ideal candidates for a differential approach. The human-based indexes of VF (see Section 2 for details) were adapted to be computed on different DSMs (Landauer & Dumais, 1997; Mikolov et al., 2013). Specifically, we adopted two predict and one count model. All three semantic spaces were based on the itWac web-crawled corpus (Baroni et al., 2009). The two predict models (Word-Embeddings Italian Semantic Space 1 and 2 - "WEISS1" and "WEISS2") were obtained from Marelli (2017) and were chosen for both their practical accessibility (http://meshugga.ugent.be/snaut-italian) and their proven good performance in previous studies (Mancuso et al., 2020; Nadalini et al., 2018). WEISS1 is based on a CBOW model with 400 dimensions and a 9-word window; WEISS2 is based on a CBOW model with 200 dimensions and a 5-word window. Both models consider words with a minimum frequency of 100 in the original corpus. The count-model based on Latent Semantic Analysis ("LSA") was created ad-hoc for this study following Günther and colleagues' (2015) procedure. Many psycholinguistic studies applying LSA in the English language used the TASA corpus (http://lsa.colorado.edu, including 12,190,931 tokens), which is a far smaller corpus than ItWac (about 1.9 billion tokens). To ensure comparability with this previous literature, we extracted a subset of the itWac corpus to match the TASA size. We selected an untagged set of 91,058 documents randomly extracted from itWAC, comprising the same set of words (N = 180,080) of the WEISS semantic spaces. The creation of a matrix of co-occurrences was carried out using the DISSECT toolkit (Dinu et al., 2013), and applying a Positive Pointwise Mutual Information weighting scheme (Niwa & Nitta, 1995), followed by dimensionality reduction by Singular Value Decomposition. We set the number of dimensions at 300 following the study of Landauer and Dumais (1997), which indicates good performance for dimensionalities ranging from 300 to 1,000.

2 Materials and Methods

The verbal production to a sematic VF (category "fruits") from the original cohort of 371 subjects (Table 1) was analysed. Overall datapoints were N = 3,642 words, with 133 unique words.

	PPA	FTD	SD	CTR
Number	16	33	15	307
Age	73.6±3.4	67.0±6.1	67.9±6.5	54.9±17
Education	7±4.6	8.6±4.4	9.3±4.9	9.6±5

Table 1: Demographic information for all the subject groups.

Data were entered in an R pipeline, leveraging on two word2vec (Mikolov et al., 2013) semantic spaces ("WEISS1" and "WEISS2"), and an LSA space with identical vocabulary size ("LSA"). For each participant, the pipeline outputs three sets of semantic indexes computed according to five different thresholds (set to identify the occurrence of a semantic switch), corresponding to the 10^{th}, 30^{th}, 50^{th}, 70^{th}, and 90^{th} quantiles of the distribution of semantic relatedness values (Table 2), computed considering the cosine proximity of all adjacent words produced by the whole study cohort.

	10^{th}	30^{th}	50^{th}	70^{th}	90^{th}
WEISS1	.185	.226	.247	.268	.287
WEISS2	.303	.371	.405	.434	.463
LSA	.336	.431	.479	.519	.582

Table 2: Cosine values adopted as thresholds for the three semantic spaces.

For each participant, we computed the following 9 indexes of VF:

1) *Total number of valid words,* produced in 1 minute, excluding repetitions. Differently from the original work, words not

included in the vocabulary of the semantic space were obligatory excluded, but words not belonging to the category "fruit" were kept. Due to limitations of the semantic space's vocabulary, 53 words and compound expressions (8 from the patient group and 45 from the control group) out of the 3,642 (1.5%) were removed from the data;

2) *Repetitions* ("rep"): the total number of repeated words;

3) *Total number of switches* ("switch"): computational equivalent of the "number of switches between subcategories" in the original work. Semantic switches were identified based on measures of semantic relatedness obtained from three semantic spaces and according to five different thresholds (Table 2);

4) *Total number of semantic clusters* ("NC"): computational equivalent of the "number of subcategories" in the original work. Clusters were identified based on the occurrence of a semantic switch, i.e., when the mean value of cosine similarity of words within a cluster drops below the identified threshold (Table 2);

5) *Mean size of clusters* ("SC"): mean number of words within a semantic cluster; computational equivalent of the "relative switching" index in the original work;

6) *Average semantic proximity* ("prox"), the semantic distance between adjacent words. Unlike the original index, based on human-derived estimated of semantic proximity (Reverberi et al., 2006), we derived this index from the mean cosine between the vectorial representation of adjacent words in the participants' production.

In addition, to ascertain the replicability of original results with computational methodologies, the following indexes were adapted from the original work:

7) *Mean familiarity* ("fam"). As a computational equivalent of the original index, calculated according to familiarity scores collected from a sample of healthy controls (Reverberi et al., 2004), we computed the raw word frequency as derived from the corpus of reference (itWac), converted to lower case and excluding metadata;

8) *Out-of-category words* ("OOC"): number of words not pertaining to the 15 subcategories of "fruit" as identified in previous works by the same Authors (Reverberi et al., 2004; 2006). Given that the vectorial representation of words differs according to inflectional morphology, data were not normalised (singular to plural) but kept as originally produced;

9) *Order Index* ("OI"): computed following the formula proposed in Reverberi et al., 2006. In its simplified notation, the Order Index is equivalent to the difference between the theoretical maximum number of switches (total number of words minus 1) and the actual observed switches, divided by the range of theoretically possible switches (total number of words minus 1, minus total number of clusters minus 1). To avoid non-linearity problems, the participant production is represented in a three-dimensional space having number of words, number of switches, and number of subcategories as axes: the order index is then transformed using the arctangents of the resulting segments.

2.1 Statistical Analyses

All variables of interest were pre-processed to remove variance due to differences in age, level of education, and the total number of words. We ran a linear regression analysis with the relevant variable as the dependent factor and with age, education, and the total number of words as regressors (only considering healthy subjects to avoid any potential bias in the estimates due to brain damage). We then used the regression coefficients to compute the residuals for each variable and all subjects. Residuals were then used as predicting variables for the classification analysis. The average for each variable and each patient group was compared with the respective average in the control group through a two-sample t-test, Bonferroni corrected.

2.2 Classification Analysis

The R packages `caret` and `e1071` (interfaces to the LIBSVM by Chang & Lin 2011) were used. The aim of the classification analysis was to determine: i) which variables, alone or in combination, would be able to classify a subject as being either a patient or control, and; ii) which variables, alone or in combination, would best classify a patient as being member of one of the three frontal dementia group (FTD, PPA, SD).

After removing variance due to differences in age and education, we performed a Leave-One-Out Cross-Validation (LOOCV) analysis. The model kernels were set as linear, and relative weights were added to counterbalance the difference in group numerosity. In LOOCV, a data instance is left out, and a model is constructed on all other data instances in the training set. The model is tested against the data point left out, and the associated error is recorded. The process is then repeated for all data points, and the overall prediction error is calculated by taking the average of the recorded test error estimates. The LOOCV analysis was repeated for each combination of the 9 variables of interest, for each of the 3 semantic spaces, and each of the 5 thresholds, resulting in 7,665 models.

3 Results

We compared the performance of each group to that of healthy controls for each of the nine variables considered. All pathological groups significantly differed from the controls on at least one variable (Table 3). In the classification analysis, we investigated which variables (alone, or in all the possible combinations with other variables, i.e., 511 combinations) would best predict the membership of participants. We carried out two sets of analysis: i) healthy controls versus participants with focal dementias (PPA, FTD, and SD); and ii) participants with PPA versus participants with FTD versus participants with SD. The analysis was performed for each semantic space and for each preidentified threshold for a total number of 7,665 models.

	FTD	PPA	SD
Proximity	+		
Familiarity			
New words	+		+
Out-Of-Category			
N Switches	+		
N Cluster	+		
Size Cluster	+	+	+
Order Index	+	+	
Repetitions			

Table 3: Variables that are significantly different between a given pathological group vis-à-vis healthy controls. Results Bonferroni-corrected for multiple comparison are reported.

The best classification performances for patients versus healthy controls was found when we considered the variables "total number of new words" and "Order Index" at any threshold and with all semantic spaces. In these cases, the overall accuracy of the models was 61.2%, with sensitivity of 57.4% and specificity of 79.7% (Table 4).

SS	Thres.	Vars	Acc.	Sens.	Spec.
Human-Based		**NC + prox + new + OOC**	**84**	**86**	**82**
all	all	New + OI	61.2	57.4	79.7
-	-	New	61.0	57.0	79.7
all	all	OI	61.0	57.0	79.7
all	all	Rep + new + OI	60.7	55.7	84.4
-	-	OOC	60.4	56.4	79.7

Table 4. Top 5 performing classification models (patients vs controls).

The best classification performances for patients in their specific pathology group was found when we considered the variables "out of category words", "average semantic proximity", and "size of clusters" computed at the 3rd threshold (50th) of the WEISS2 space (Table 5). In this case, the overall max accuracy was 43.8%. Sensitivity and specificity for each pathology group were: PPA = 87.5% and 62.5%; FTD = 36.4% and 71%; SD = 13.33% and 81.6%, respectively.

SS	Thres.	Vars	Acc.	PPA	FTD	SD
Human-Based		**Fam + NS + OI + new + rep**	**58**	**NA**	**NA**	**NA**
W2	50	OOC + prox + SC	43.8	87.5/ 62.5	36.4/ 71	13.3/ 81.6
W1	10	OOC + SC	42.2	87.5/ 56.3	39.4/ 74.2	0/ 83.7
W1	30	NS + NC	40.6	93.8/ 50	33.3/ 77.4	0/ 85.7
W1	70	OOC + SC	40.6	87.5/ 62.5	36.4/ 64.5	0/ 81.6
W2	90	SC	39.1	68.8/ 60.4	42.4/ 64.5	0/81. 6

Table 5. Top 5 performing classification models (patients in each specific pathology group).

4 Discussion

In this work, we replaced human-based measures of semantic proximity with DSM-derived measures of semantic proximity to compute a set of indexes of VF that was found to be able to classify with good accuracy people with and without focal dementias based on their verbal production to a semantic VF task (category "fruits", which was originally adopted to limit the set of possible items as compared to broader categories such as “animals”). The objective of the study was to assess the accuracy of Machine Learning (ML) models based on DSM measures of semantic information, in view of their possible extension to words and semantic categories for whom the measure of semantic proximity is not available. Despite being above chance in both cases, ML models based on DSM-derived measures of semantic proximity showed lower accuracy compared to models built on human-based ratings. This was true both for the classification of patients versus controls (61.2% and 84%, respectively), as well as for the subclassification of diagnosis (43.8% and 58%, respectively).

The observed differences might be due to the functional adaptations needed to transpose the original VF indexes to DSM-derived measures. For example, the computational equivalent of the "familiarity" index, calculated according to familiarity scores collected from the sample of healthy controls, was approximated via the raw word frequency as derived from the corpus of reference. Moreover, given that the vectorial representation of words differs according to inflectional morphology, data were not normalised (singular to plural) but kept as originally produced, unlike the original work. Hence, it might be possible that these operations introduced some distortions that could explain the differences observed compared to the original study.

In terms of parameter setting, it is worth noting that our choices might have affect the overall performance of the adopted models, possibly reducing their ability to avoid noise and biases. For example, according to Tripodi (2017), hyperparameter setting for Italian has specific requirements in terms of vector size, negative sampling, vocabulary threshold cutting, to maximize performance in an analogy task (although to what extent such recommendation can be extended to VF is an empirical question that remains to be addressed). Also, the choice of a CBOW model, instead of “more predictive” algorithms such as Skipgram and Mask might have reduced the ability of the model to mimic the human ratings of word associations.

However, a different explanation might be related to the type of information encoded into the human proximity ratings. Given its evolutionary relevance, the neural substrate underpinning the notion of "fruits" might encode a rich multidimensional semantic characterisation (including sensory information such as taste, smell, sight, touch). As such, the representation of this semantic category might not be simply derivable by the lexical distribution of its items in a corpus. Differently, other semantic categories might leverage on less perceptual and more encyclopaedic semantic knowledge, such as, for example, the category "animals", another semantic cue widely used for the assessment of VF. Indeed, while people do generally have first-hand, real-life experience of "fruits", knowledge about "animals" may be more commonly derived from indirect exposure to encyclopaedic information (i.e., the media). In other words, when we think about a cherry, we may not only recall the meaning of the lemma as compared to, for example, an apple, but at the same time, we might also recall the sensory information attached to the drupe (round, red, juicy, etc.). Conversely, apart from common pets, it is unlikely that participants have first-hand experience about most of the items commonly included "animals" category (e.g., "lion", “whale”, etc.).

This means that distributional models might be not the best-suited tool to resolve semantic problems when the semantic task under investigation makes use of a subset of words pertaining to a semantic category perceptually rich (such as that of “fruits”).

5 Conclusions and Future Works

The past decades have witnessed an increasing interest towards the application of NLP techniques to answer, or support the resolution of, different clinical problems, from patients’ classifications to disease monitoring, and from differential diagnosis to prediction of treatment response (see de Boer et al., 2018 for a comprehensive review). All these applications implicitly rely on the assumption that these techniques are agnostic/transparent to the semantic task under investigation and, given the good results obtained, that they are equipped with sufficiently rich semantic information to solve any kind of task based on linguistic data. Our findings challenge this idea and align with previous works pointing to a lack of basic features of perceptual meaning in DSM (Lucy and Gauthier, 2017).

Implications for the application of DSM-derived measures to clinical work and research indicate that the choice of the verbal task and the associated DSM can affect the results. For this reason, we plan to assess the classification accuracy of ML models built both on human ratings and DSM-derived measures of semantic proximity for other categorical VF tasks, as well as adopting word vectors derived from lemmatised corpora.

Before moving to more recent language models such as the last generation of deep neural language models like BERT (Devlin et al., 2019), consideration should be given to the trade-off between computational and data resources needed to train them (Bender et al., 2021) on one hand, and what kind of added value they can give compared to traditional "static" embeddings (Lenci et al., 2021) on the other. Further research might address the limits of current DSM models by enriching the information encoded, integrating experiential and distributional data to induce reliable semantic representations (Andrews et al., 2009). Additional sources of multimodal information (e.g., Lynnott et al., 2020) including visual and audio information, might help overcome these current limitations (Chen et al., 2021).

References

Baroni Marco, Bernardini Silvia, Ferraresi Adriano and Zanchetta Eros. 2009. The waCky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3): 209–226.

Bender Emily M., Gebru Timnit, McMillan-Major Angelina & Shmitchell Shmargaret. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?🦜. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*: 610-623.

Chang Chih-Chung and Lin Chih-Jen. 2011. LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology* (TIST), 2(3): 1-27.

Chen Wei, Wang Weiping, Liu Li and Lew Micheal S. 2021. New ideas and trends in deep multimodal content understanding: A review. *Neurocomputing, 426*:195-215.

De Boer Jann N., Voppel Alban E., Begemann Marieke J.H., Schnack Hugo G., Wijnen Frank and Sommer Iris E.C. 2018. Clinical use of semantic space models in psychiatry and neurology: a systematic review and meta-analysis. *Neuroscience & Biobehavioral Reviews, 93:* 85-92.

Devlin Jacob, Chang MW, Lee K, Toutanova K. 2019 BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: *Proceedings of NAACLHLT 2019*, 4171–4186

Dinu Georgiana and Baroni Marco. 2013. Dissect-distributional semantics composition toolkit. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 31-36.

Günther Fritz, Dudschig Caroline and Kaup Barbara. 2015. Latent semantic analysis cosines as a cognitive similarity measure: Evidence from priming studies. *Quarterly Journal of Experimental Psychology, 69*(4):626–653.

Landauer Thomas and Dumais Susan. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2), 211.

Lenci Alessandro, Sahlgren Magnus, Jeuniaux Patrick, Gyllensten Amaru Cuba and Miliani Martina 2021. A comprehensive comparative evaluation and analysis of Distributional Semantic Models. arXiv preprint arXiv:2105.09825.

Lezak Muriel, Howieson Diane, Loring David, Hannay Julia and Fischer Jill. 2004. *Neuropsychological assessment*. New York: OUP, USA.

Lucy Li and Gauthier Jon. 2017. Are Distributional Representations Ready for the Real World? Evaluating Word Vectors for Grounded Perceptual Meaning. *Proceedings of the First Workshop on Language Grounding for Robotics.*

Lynott Dermot, Connell Louise, Brysbaert Marc, Brand James and Carney James. 2020. The Lancaster Sensorimotor Norms: multidimensional measures of perceptual and action strength for 40,000 English words. *Behavior Research Methods*, 52(3), 1271-1291.

Mandera Paul, Keuleers Emmanuel and Brysbaert Marc. 2017. Explaining human performance in psycholinguistic tasks with models of semantic similarity based on prediction and counting: A review and empirical validation. *Journal of Memory and Language*, 92, 57-78.

Marelli Marco. 2017. Word-embeddings Italian Semantic spaces: A semantic model for psycholinguistic research. *Psihologija*, 50(4): 503–520.

Mikolov Tomas, Sutskever Ilya, Chen Kai, Corrado Greg and Dean Jeffrey. 2013. Distributed Representations of Words and Phrases and their Compositionality. Retrieved from http://arxiv.org/abs/1310.4546

Niwa Yoshiki and Nitta Yoshihiko. 1995. Co-occurrence vectors from corpora vs. distance vectors from dictionaries. arXiv preprint cmp-lg/9503025

R CoreTeam. 2021. R: A language and environment for statistical computing. R Foundation for Statistical Computing. Retrieved from https://www.r-project.org.

Reverberi Carlo, Cherubini Paolo, Baldinelli Sara and Luzzi Simona. 2014. Semantic fluency: Cognitive basis and diagnostic performance in focal dementias and Alzheimer's disease. *Cortex*, 54, 150-164.

Tripodi Rocco and Pira Stefano Li. 2017. Analysis of Italian word embeddings. arXiv preprint arXiv:1707.08783.

Turney Peter D. and Pantel Patrick. 2010. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37, 141-188

WITS: Wikipedia for Italian Text Summarization

Silvia Casola[1,2], **Alberto Lavelli**[2]
1. Università degli studi di Padova
2. Fondazione Bruno Kessler
scasola@fbk.eu, lavelli@fbk.eu

Abstract

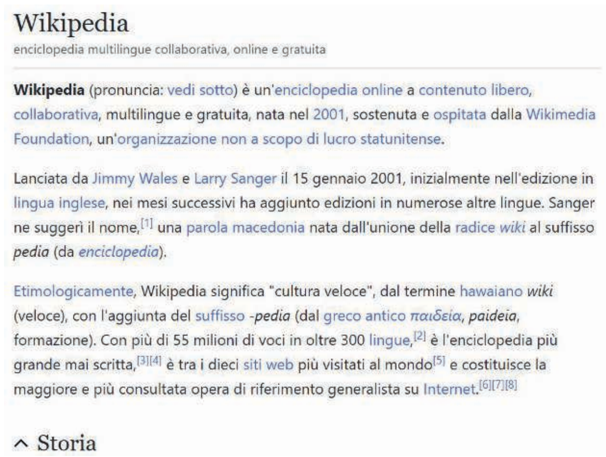
Abstractive text summarization has recently improved its performance due to the use of sequence to sequence models. However, while these models are extremely data-hungry, datasets in languages other than English are few. In this work, we introduce WITS (Wikipedia for Italian Text Summarization), a large-scale dataset built exploiting Wikipedia articles' structure. WITS contains almost 700,000 Wikipedia articles, together with their human-written summaries. Compared to existing data for text summarization in Italian, WITS is more than an order of magnitude larger and more challenging given its lengthy sources. We explore WITS characteristics and present some baselines for future work.

1 Introduction

Automatic text summarization aims at condensing one or more source documents in a shorter output, which contains their most salient information. The underlying task can be framed in two different manners: extractive summarizers select the most relevant segments from the input and produce a summary which is a concatenation of such segments; as a result, the output is a subset of the original text, which the summary follows verbatim. On the other hand, abstractive summarizers aim to encode the whole source into an internal representation from which they generate the summary; thus, they produce a new piece of text that condenses the source without necessarily using its vocabulary and expressions.

Recently, abstractive summarization has attracted a growing interest in the Natural Language Processing (NLP) community. Sequence to sequence models have been increasingly used for the task, with pre-trained encoder-decoder transformers becoming the de facto state of the art for abstractive text summarization. Normally pre-trained in an unsupervised manner, these models are then fine-tuned in a supervised way on the downstream dataset; during fine-tuning, the model learns to generate the summary from the source document.

While various datasets for abstractive summarization exist for English, resources in other languages are limited. This paper introduces WITS (Wikipedia for Italian Text Summarization), a large-scale dataset for abstractive summarization in Italian, built exploiting Wikipedia. Taking advantage of the structure of Wikipedia pages, which contain a lead section (Figure 1) – giving an overview of the article's topic –, followed by the full-length article – describing the topic in details –, we create a large and challenging dataset for abstractive summarization in Italian, which we will


Wikipedia
enciclopedia multilingue collaborativa, online e gratuita

Wikipedia (pronuncia: vedi sotto) è un'enciclopedia online a contenuto libero, collaborativa, multilingue e gratuita, nata nel 2001, sostenuta e ospitata dalla Wikimedia Foundation, un'organizzazione non a scopo di lucro statunitense.

Lanciata da Jimmy Wales e Larry Sanger il 15 gennaio 2001, inizialmente nell'edizione in lingua inglese, nei mesi successivi ha aggiunto edizioni in numerose altre lingue. Sanger ne suggerì il nome,[1] una parola macedonia nata dall'unione della radice *wiki* al suffisso *pedia* (da *enciclopedia*).

Etimologicamente, Wikipedia significa "cultura veloce", dal termine hawaiano *wiki* (veloce), con l'aggiunta del suffisso *-pedia* (dal greco antico *παιδεία*, *paideia*, formazione). Con più di 55 milioni di voci in oltre 300 lingue,[2] è l'enciclopedia più grande mai scritta,[3][4] è tra i dieci siti web più visitati al mondo[5] e costituisce la maggiore e più consultata opera di riferimento generalista su Internet.[6][7][8]

^ Storia

Figure 1: The lead section (from the Wikipedia' own page), which we consider as the article summary. We use the remaining of the article as the source.

make publicly available.

WITS is particularly challenging, given its large source length and its high abstractiveness. In this paper, we describe the dataset, its statistics and characteristics, and report some preliminary experiments that might be used as baselines for future work.

This paper is organized as follows: in Section 2, we describe the state of the art in text summarization, focusing on resources for Italian. We later preset the dataset and its related task (Section 3.1); we describe the data collection and preprocessing process in Sections 3.2 and 3.3. In Section 4, we show our results when summarising the dataset using some existing extractive baseline models. Finally, we draw our conclusions in Section 5.

2 State of the Art

Automatic text summarization has recently attracted increasing attention from the NLP community. However, the majority of the research work still focuses on English.

As a matter of example, out of all the papers published in the Association for Computational Linguistics (ACL) conference in 2021, 46 explicitly refer to summarization in their title; 38 of these dealt with English only, while 7 presented experiments with one or more other languages (including 2 on source code summarization). For reference, only one paper (Mastronardo and Tamburini, 2019) on text summarization (in English) was published at the Italian Conference on Computational Linguistics (CLiC-it) since its first edition, and none experimented with Italian.

In this section, we present the state of the art in abstractive text summarization. We first present the available datasets for the task; then, we discuss some relevant learning models. We focus on the significant gap between English and Italian, for which very few resources exist.

2.1 Datasets for Automatic Text Summarization

A typical dataset for text summarization is composed of some source documents (which needs to be summarized) and their corresponding summaries, used as the gold standard. A minority of datasets (e.g., the DUC 2004 dataset[1]) provide multiple gold standards; however, such datasets tend to be small and are mostly used for evaluation.

In general, summaries exploit a human-written abstract. For example, the CNN/Daily Mail Corpus (Nallapati et al., 2016)[2] leverages a bullet-point summary on the newspapers' websites. A similar rationale is used in datasets constructed from scientific papers (Cohan et al., 2018)[3] or patents (Sharma et al., 2019)[4]. In contrast, Rush et al. (2015)[5] frames the task of news summarization as headline generation.

To the best of our knowledge, WikiLingua (Ladhak et al., 2020)[6] is the only summarization dataset that contains data in Italian. WikiLingua is a cross-lingual dataset for abstractive text summarization built on top of WikiHow. WikiHow contains tutorials on how to perform specific tasks in the form of step-by-step instructions. The dataset constructs a summary by concatenating the first sentence for each step and using the remaining text as the source. WikiLingua contains data in 18 languages, including Italian (50,943 source-summary pairs). Both summaries and sources are relatively short (on average, 44 and 418 tokens, respectively, for the Italian split).

2.2 Models for Abstractive Text Summarization

Abstractive text summarization is one of the most challenging tasks in NLP: it requires very long input understanding (encoding), salient passages finding and constrained text generation. Technically, models for abstractive text summarization are generally sequence-to-sequence: they encode the input and then generate the output through a neural network. While some previous work used Recurrent Neural Networks (Chung et al., 2014), with the possible addition of an encoder-decoder attention mechanism (Chopra et al., 2016), transformer models (Vaswani et al., 2017) have later become pervasive, following a similar trend in many other NLP areas. Using self-attention, these models have proved to be superior to Recurrent

[1] https://duc.nist.gov/duc2004/

[2] https://huggingface.co/datasets/cnn_dailymail

[3] https://huggingface.co/datasets/arxiv_dataset

[4] https://huggingface.co/datasets/big_patent

[5] https://huggingface.co/datasets/gigaword

[6] https://huggingface.co/datasets/wiki_lingua

Neural Networks, as they are able to better deal with long dependencies, a critical task in text summarization.

Following another recent trend in NLP, many summarization models use a transfer-learning approach: after a pre-training phase, in which they are training in an unsupervised way on a huge amount of text, they are fine-tuned for the specific downstream task on a relatively limited amount of supervised data. Summarization models either exploit encoders and decoders previously trained for other tasks or are pre-trained from scratch on a specific objective tailored for summarization. Rothe et al. (2020), for example, leveraged previously existing pre-trained models (BERT in Devlin et al. (2019); ROBERTA in Liu et al. (2019); and GPT-2[7] in Radford et al. (2019)) as encoders or decoders of the sequence-to-sequence summarizer and showed high performance improvement with respect to random initialization. More recently, summarization models (Song et al., 2019; Lewis et al., 2020) have been pre-trained with an objective specific to Natural Language Generation tasks. For example, authors of Pegasus (Zhang et al., 2020) used two objectives: Masked Language Model (Devlin et al., 2019) has been widely used in previous work, and consists in masking a percentage of tokens in text, later predicted using context; Gap Sentences Generation is instead a new pre-training objective, in which a percentage of the original sentences are masked, and the model needs to generate them in accordance to the contest.

Following a shared practice, most summarization models have first been trained and evaluated for English only. In some cases, a subsequent multilingual version of the model was also created (Xue et al., 2021). To the best of our knowledge, few sequence-to-sequence models in Italian exist to date[8], and while they might be fine-tuned for summarization, no full-scale evaluation has been performed yet.

[7]GPT-2 ha also been adapted for Italian. See: De Mattei, L., Cafagna, M., Dell'Orletta, F., Nissim, M., & Guerini, M. 2020. GePpeTto Carves Italian into a Language Model. In CLiC-it 2020

[8]See, for example, IT5-base (`https://huggingface.co/gsarti/it5-base`)

3 WITS

3.1 Task and Rationale

Given a Wikipedia article, we extract the lead section (which we sometimes refer to as ”Summary” in the remaining of the paper) and propose the following task:

> Given all article sections, summarize its content to produce its lead section.

The task is rather natural given pages structure. According to the Wikipedia Manual of Style[9], the lead section is, in fact, a high-quality summary of the body of the article. The lead “serves as an introduction to the article and a summary of its most important contents” and “gives the basics in a nutshell and cultivates interest in reading on—though not by teasing the reader or hinting at what follows”. Moreover, it should “stand on its own as a concise overview of the article's topic”.

As for the content, according to Wikipedia, the lead must define the topic, explaining its importance and the relevant context; then, it must summarize the most prominent points of the article, emphasizing the most important material.

Moreover, the lead should only cover information that is contained in the article: “significant information should not appear in the lead if it is not covered in the remainder of the article”. This is particularly relevant for abstractive summarization, as models are more prone to produce summaries that are not factual to the source (often called hallucinations) when they are trained to generate summaries containing information not in the source (Nan et al., 2021). The problem of factuality in abstractive summarization is currently an active area of research, as previous work has shown that up to 30% of generated summaries contain non-factual information (Cao et al., 2018).

Linguistically, the lead “should be written in a clear, accessible style with a neutral point of view”. It is worth noting that, in contrast to WikiLingua, where the summary is constructed as a concatenation of sentences from different parts of the articles, the summary in WITS is a stand-alone piece of text, with a coherent discourse structure.

3.2 Data Collection

This section describes the process of data collection and preprocessing.

[9]`https://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style`

	WITS		IT-Wikilingua	
# docs	699,426		50,943	
	Summary	Source	Summary	Source
# sentences (avg)	3.75	33.33	5.01	23.52
# tokens (avg)	70.93	956.66	23.52	418.6
Comp. ratio (avg)	16.14		11.67	

Table 1: Datasets statistics. spacy is used for text and sentence tokenization. The number of tokens and sentences is computed for all documents and then averaged.

	WITS		IT-Wikilingua	
	Summary	Source	Summary	Source
PER (avg)	1.13	26.21	0.32	1.05
LOC (avg)	2.03	24.07	0.42	1.39
ORG (avg)	0.60	6.65	0.68	0.37
MISC (avg)	19.68	19.68	0.84	3.07
All (avg)	23.44	76.61	1.65	5.88

Table 2: Named Entities in WITS and IT-WikiLingua.

We downloaded the latest XML dump of Wikipedia in Italian[10], which contains text only. We used Python and the Gensim library to process the file[11]. The original number of documents was 1,454,884. We applied the following exclusion criteria: we removed pages whose title contains numbers only (as they mostly describe years and contain lists of events and references), lists (titles starting with "Lista d"), pages with summaries with less than 80 characters and articles and pages for which the article is less than 1.5 times longer than the lead.

We then preprocessed the text in the following way: from the summary, we removed the content of parentheses (as they often contain alternative names or names in a different language, which cannot be inferred from the article). For the article, we further excluded the following sections, which are not relevant for our task: Note (Footnotes), Bibliografia (References), Voci correlate (See also), Altri progetti (Other projects), Collegamenti esterni (External links), Galleria di Immagini (Images).

3.3 Dataset Statistics

Table 1 shows some statistics on the dataset and compares WITS with the Italian split of WikiLingua (which we will refer to as IT-WikiLingua).

IT-WikiLingua contains documents from 17,673 WikiHow pages, but some of these pages describe more than one method related to the same topic. For example, the page "How to Reduce the Redness of Sunburn" contains several methods: "Healing and Concealing Sunburns", "Lessening Your Pain and Discomfort", and "Preventing a Sunburn". We consider distinct methods as separate documents, as they can be summarized in isolation. Notice that WITS is more than an order of magnitude larger than IT-Wikilingua.

We computed the number of tokens and the number of sentences through the spaCy it-core-news-lg[12] model. Compared to IT-WikiLingua, documents in WITS contains more tokens both in their summary and in their source (which is more than double in length), making the dataset particularly challenging. Note how the sentences are also more lengthy (thus complex) on average. For example, summaries in WITS contain on average less than 4 sentences, but more than 70 words; in contrast, IT-WikiLingua's summaries consist of more than 5 sentences but contain on average 44 tokens. Not surprisingly, WITS' compression ratio is larger than IT-WikiLingua's and very high in absolute value. Finally, we also notice that the dataset is very rich in named entities. Table 2 reports the Named Entities as extracted with spaCy from WITS and IT-Wikilingua.

4 Baselines

We tested some preliminary non-neural baseline methods on the dataset, reported in Table 3.

All methods reported are unsupervised. Thus, we unsupervisedly obtained the summary from the source and then used the lead as the gold standard for evaluation. We evaluated the summaries using Recall-Oriented Understudy for Gisting Evaluation (ROUGE) (Lin, 2004). ROUGE is an n-gram based, recall-oriented metric for summary quality evaluation. Following previous work (Lloret et al., 2018), we report ROUGE-1 (R-1), ROUGE-2 (R-2), and ROUGE-L (R-L) (recall).

We considered the following baselines:

Lead-3 We extract the first three sentences from the source. Previous work has shown that this baseline is often hard to beat (See et al., 2017), especially in news summarization,

[10] https://dumps.wikimedia.org/itwiki/latest/itwiki-latest-pages-articles.xml.bz2

[11] https://radimrehurek.com/gensim/scripts/segment_wiki.html

[12] https://spacy.io/models/it

which presents an "inverted pyramid" structure and tends to report the most important content at the start.

TextRank (Mihalcea and Tarau, 2004) TextRank is an unsupervised algorithm that extracts the most relevant sentences in the source. The algorithm constructs a graph with sentences as nodes and sentence similarity (in terms of shared vocabulary) as edges. The sentences are then ranked by using the PageRank (Page et al., 1999) algorithm.

LexRank (Erkan and Radev, 2004) LexRank works in a similar way as TextRank. However, instead of computing sentence similarity on normalized shared vocabulary, it uses the cosine similarity of their TF-IDF vectors.

SumBasic (Nenkova and Vanderwende, 2005) SumBasic extracts sentences based on their word probabilities. Specifically, it scores each sentence as the mean of the probability of the words it contains (based on their frequency in the document). Iteratively, the sentence with the best score among the ones containing the most probable word is chosen. The probability of the words in the chosen sentence is then squared to limit redundancy.

IT5-small (Raffel et al., 2020) The Text-to-Text Transfer Transformer (T5) is a pre-trained sequence-to-sequence language model, trained treating both input and output as text strings; the rationale is to use the same models for all NLP tasks, unifying them under the sequence-to-sequence framework. We use a small version of the original model (60 million parameters)[13], pretrained on the Clean Italian mC4 IT[14], the Italian split of the multilingual cleaned version of Common Crawl's Corpus (mC4) (Raffel et al., 2020). We extracted 10,000 summary-source pairs from the dataset for the validation set, and 10,000 for the test set. We trained the model on the rest of the data for 100,000 steps; this account for around 30% of the training data. We trained on two GeForce RTX 2080 GPUs and kept the batch size per GPU to 1. We kept the summary length to 75 tokens, and the source text length to 1000 tokens.

	R-1	R-2	R-L
Lead-3	24.76	5.54	16.54
TextRank	30.20	6.57	19.67
LexRank	26.90	5.91	17.52
SumBasic	20.60	4.80	14.01
IT5-small	21.58	9.69	19.34

Table 3: ROUGE results on WITS.

Results show that the Lead-3 baseline performance is low; this is likely due to the structure of Wikipedia, which contains several thematic sections without a general introduction outside the lead section. Extracting the first sentence(s) from each section would likely produce better results and could be investigated in future work.

In contrast, TextRank is the best non-neural baseline, with a ROUGE-2 score of 6.57; LexRank performs comparably. SumBasic metrics are even lower than those obtained with the Lead-3 baseline, suggesting that a purely frequency-based approach is insufficient given the dataset complexity.

Finally, the neural baseline achieves the best results in terms of ROUGE-2, even if it is relatively small and likely severely under-trained, since only around 30% of the data are used for fine-tuning, due to computational constraints. This suggests that sequence-to-sequence neural models have great potential on the dataset, and should be better investigated in future work. Surprisingly, however, results in terms of ROUGE-1 are instead below most of the other baselines. Future work should investigate this discrepancy.

5 Conclusions

We have presented WITS, the first large-scale dataset for abstractive summarization in Italian. We have exploited Wikipedia's articles' structure to build a challenging, non-technical dataset, with high-quality human-written abstracts. Given the lengthy source documents, the short summaries and the short extractive fragments, the dataset calls for an abstractive approach. In the paper, we have explored some standard non-neural extractive baselines and a neural abstractive baseline. Future work will investigate further neural baselines for

[13] https://huggingface.co/gsarti/it5-small

[14] https://huggingface.co/datasets/gsarti/clean_mc4_it

the dataset. Moreover, the dataset can be easily extended applying the procedure described in the paper to more languages, including low-resource ones given Wikipedia structure. We are confident that research in summarization in languages other than English will become more active in the near future and hope that WITS can be a valuable step in this direction.

References

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: Fact aware neural abstractive summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Sumit Chopra, Michael Auli, and Alexander M. Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98, San Diego, California, June. Association for Computational Linguistics.

Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana, June. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Günes Erkan and Dragomir R. Radev. 2004. LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22(1):457–479, December.

Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A new benchmark dataset for multilingual abstractive summarization. In *Findings of EMNLP, 2020*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *ArXiv*, abs/1907.11692.

Elena Lloret, Laura Plaza, and Ahmet Aker. 2018. The Challenging Task of Summary Evaluation: An Overview. *Language Resources and Evaluation*, 52(1).

C. Mastronardo and F. Tamburini. 2019. Enhancing a text summarization system with ELMo. In *CLiC-it*.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain, July. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany, August. Association for Computational Linguistics.

Feng Nan, Ramesh Nallapati, Zhiguo Wang, Cicero Nogueira dos Santos, Henghui Zhu, Dejiao Zhang, Kathleen McKeown, and Bing Xiang. 2021. Entity-level factual consistency of abstractive text summarization. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2727–2733, Online, April. Association for Computational Linguistics.

Ani Nenkova and Lucy Vanderwende. 2005. The impact of frequency on summarization. Technical report, Microsoft Research.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2020. Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, 8:264–280.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal, September. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada, July. Association for Computational Linguistics.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy, July. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: masked sequence to sequence pre-training for language generation. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 5926–5936. PMLR.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *NAACL*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.

Policycorpus XL: An Italian Corpus for the Detection of Hate Speech Against Politics

Fabio Celli[1], **Mirko Lai**[2], **Armend Duzha**[1], **Cristina Bosco**[2], **Viviana Patti**[2]
1. Research & Development, Gruppo Maggioli, Italy
2. Dept. of Informatics, University of Turin, Italy
fabio.celli@maggioli.it, mirko.lai@unito.it,
armend.duzha@maggioli.it, bosco@di.unito.it, patti@di.unito.it

Abstract

In this paper we describe the largest corpus annotated with hate speech in the political domain in Italian. Policycorpus XL has 7000 tweets, manually annotated, and a presence of hate labels above 40%, while in other corpora of the same type is usually below 30%. Here we describe the collection of data and test some baseline with simple classification algorithms, obtaining promising results. We suggest that the high amount of hate labels boosts the performance of classifiers, and we plan to release the dataset in a future evaluation campaign.

1 Introduction and Background

In recent years, computer mediated communication on social media and microblogging websites has become more and more aggressive (Watanabe et al., 2018). It is well known that people use social media like Twitter for a variety of purposes like keeping in touch with friends, raising the visibility of their interests, gathering useful information, seeking help and release stress (Zhao and Rosson, 2009), but the spread of fake news (Shu et al., 2019; Alam et al., 2016) has exacerbated a cultural clash between social classes that emerged at least since after the debate about Brexit (Celli et al., 2016) and more recently during the pandemics (Oliver et al., 2020). Despite the fact that the behavior online is different from the behavior offline (Celli and Polonio, 2015), we observe more and more hate speech in social media, to the point where it has become a serious problem for free speech and social cohesion.

Hate speech is defined as any expression *that is abusive, insulting, intimidating, harassing, and/or incites, supports and facilitates violence, hatred, or discrimination. It is directed against people (individuals or groups) on the basis of their race, ethnic origin, religion, gender, age, physical condition, disability, sexual orientation, political conviction, and so forth* (Erjavec and Kovačič, 2012). In response to the growing number of hate messages, the Natural language Processing (NLP) community focused on the classification of hate speech (Badjatiya et al., 2017) and the analysis of online debates (Celli et al., 2014). In particular, many worked on systems to detect offensive language against specific vulnerable groups (e.g., immigrants, LGBTQ communities among others) (Poletto et al., 2017) (Poletto et al., 2021), as well as aggressive language against women (Saha et al., 2018). An under-researched - yet important - area of investigation is anti-politics hate: the hate speech against politicians, policy makers and laws at any level (national, regional and local). While anti-policy hate speech has been addressed in Arabic (Guellil et al., 2020) and German (Jaki and De Smedt, 2019), most European languages have been under-researched. The bottleneck in this field of research is the availability of data to train good hate speech detection models. In recent years, scientific research contributed to the automatic detection of hate speech from text with datasets annotated with hate labels, aggressiveness, offensiveness, and other related dimensions (Sanguinetti et al., 2018). Scholars have presented systems for the detection of hate speech in social media focused on specific targets, such as immigrants (Del Vigna et al., 2017), and language domains, such as racism (Kwok and Wang, 2013), misogyny (Basile et al., 2019) or cyberbullying (Menini et al., 2019). Each type of hate speech has its own vocabulary and its own dynamics, thus the selection of a specific domain is crucial to obtain clean data and to restrict the scope of experiments and learning tasks.

In this paper we present a new corpus, called Policycorpus XL, for hate speech detection from Twitter in Italian. This corpus is an extension of the Policycorpus (Duzha et al., 2021). We selected Twitter as the source of data and Italian as the target language because Italy has, at least since the elections in 2018, a large audience that pays attention to hyper-partisan sources on Twitter that are prone to produce and retweet messages of hate against policy making (Giglietto et al., 2019).

The paper is structured as follows: after a literature review (Section 2), we describe how we collected and annotated the data (Section 3), we evaluate some baselines (Section 4), and we pave the way for future work (Section 5).

2 Related Work

Hate Speech in social media is a complex phenomenon, whose detection has recently gained significant traction in the Natural Language Processing community, as attested by several recent review works (Poletto et al., 2021). High-quality annotated corpora and benchmarks are key resources for hate speech detection and haters profiling in general (Jain et al., 2021), considering the vast number of supervised approaches that have been proposed (MacAvaney et al., 2019).

Early datasets on Hate Speech, especially in English, were produced outside any evaluation campaigns (Waseem and Hovy, 2016), (Founta et al., 2018) as well as inside such competitions. These include SemEval 2019, where a multilingual hate speech corpus against immigrants and women in English and Spanish (Basile et al., 2019) was released, and PAN 2021, that provided a dataset for the detection of hate spreader authors in English and Spanish (Rangel et al., 2021). Most Italian datasets in the field of hate speech have been released during competitions and evaluation campaigns. There are:

- the Italian HS corpus (Poletto et al., 2017),
- HaSpeeDe-tw2018 and HaSpeeDe-tw2020, the datasets released during the EVALITA campaigns (Sanguinetti et al., 2020),
- the Policycorpus (Duzha et al., 2021), the only dataset in Italian that is annotated with hate speech in the political domain.

The Italian HS corpus is a collection of more than 5700 tweets manually annotated with hate speech, aggressiveness, irony and other forms of potentially harassing communication. The HaSpeeDe-tw corpora are two collections of 4000 and 8100 tweets respectively, manually annotated with hate speech labels and containing mainly anti-immigration hate (Bosco et al., 2018). The Policycorpus is a collection of 1260 tweets manually annotated with hate speech labels against politics and politicians. We decided to expand it and produce a new dataset.

Hate speech is hard to annotate and hard to model, with the risk of creating data that is biased and making the models prone to overfitting. In addition to this, literature also reports cases of annotators' insensitivity to differences in dialect that can lead to racial bias in automatic hate speech detection models, potentially amplifying harm against minority populations. It is the case of African American English (Sap et al., 2019) but it potentially applies to Italian as well, as it is a language full of dialects and regional offenses.

Hate speech is intrinsically associated to relationships between groups, and also relying in language nuances. There are many definitions of hate speech from different sources, such as European Union Commission, International minorities associations (ILGA) and social media policies (Fortuna and Nunes, 2018). In most definitions, hate speech has specific targets based on specific characteristics of groups. Hate speech is to incite violence, usually towards a minority. Moreover, hate speech is to attack or diminish. Additionally, humour has a specific status in hate speech, and it makes more difficult to understand the boundaries about what is hate and what is not.

In the political domain we find all of these aspects, especially messages against a minority (politicians) to attack or diminish. We think that more resources are needed for the classification of hate speech in Italian in the political domain, hence we decided to collect and annotate more data for this task.

In the next section, we describe how we created the dataset and annotated it with hate speech labels.

3 Data Collection and Annotation

Starting from the Policycorpus, we expanded it from 1260 to 7000 tweets in Italian, collected using snowball sampling from Twitter APIs. As initial seeds, we used the same set of hashtags used for the Policycorpus, for instance: #dpcm (decree of the president of the council of ministers), #legge (law) and #leggedibilancio (budget law). We removed duplicates, retweets and tweets containing only hashtags and urls. At the end of the sampling process, the list of seeds included about 6000 hashtags that co-occurred with the initial ones. We grouped the hashtags into the following categories:

- Laws, such as #decretorilancio (#relaunchdecree), #leggelettorale (#electorallaw), #decretosicurezza (#securitydecree)
- Politicians and policy makers, such as #Salvini, #decretoSalvini (#Salvinidecree), #Renzi, #Meloni, #DraghiPremier
- Political parties, such as #lega (#league), #pd (#Democratic Party)
- Political tv shows, such as #ottoemezzo, #nonelarena, #noneladurso, #Piazzapulita
- Topics of the public debate, such as #COVID, #precari (#precariousworkers), #sicurezza (#security), #giustizia (#justice), #ItalExit
- Hyper-partisan slogans, such as #vergognaConte (#shameonConte), #contedimettiti (#ConteResign) or #noicontrosalvini (#WeareagainstSalvini)

Examples of collected hashtags are reported in Figure 1

Recent shared tasks (Agerri et al., 2021; Cignarella et al., 2020; Aker et al., 2016) promoted the use of contextual information about the tweet and its author (including his/her social media network) for improving the performance of stance detection. Here, with the aim to stimulate the exploration of data augmentation on hate speech detection, we shared additional contextual information based on the post such as: the number of retweets and the number of favours (the number of tweets that given user has marked as favorite - favours_count field) the tweet received, the device used for posting it (e.g. iOS or Android), the posting date and location, and an attribute that states if the post is a tweet, a retweet, a reply, or a quote. Furthermore, we collected contextual information related to the authors of these posts such as: the

Figure 1: Wordclouds of the hashtags collected with frequency higher than 2.

number of tweets ever posted, the user's description and location, the number of her/his followers and of her/his friends, the number of public lists that this user is a member of and the date her/his account has been created.

All these contextual information are respectively part of the "root-level" attributes of the *Tweets* and *Users* objects that Twitter returns in JSON format through its APIs. Additionally, we planned to explore the interests of the author collecting the list of her/his following (the users she/he follows) employing the *following* API endpoint. Moreover, for exploring the author's social interactions, we used the *Academic Full Search API* for recovering the list of the users that she/he has retweeted to and replied to in the last two years.

The enhanced Policycorpus has been finally anonymised mapping each *tweet_id*, *users_id*, and *mention* with a randomly generated ID. To produce gold standard labels, we asked two Italian native speakers, experts of communication, to manually label the tweets in the corpus, distinguishing between hate and normal tweets according to the following guidelines: By definition, hate speech is *any expression that is abusive, insulting, intimidating, harassing, and/or incites to violence, ha-*

tred, or discrimination. It is directed against people on the basis of their race, ethnic origin, religion, gender, age, physical condition, disability, sexual orientation, political conviction, and so forth. (Erjavec and Kovačič, 2012). Below We provide some examples with translation in English:

1. "*Un chiaro #NO all #Olanda che ci vorrebbe sì utilizzatori delle risorse economiche del #MES ma in cambio della rinuncia dell Italia alla propria autonomia di bilancio. All Olanda diciamo: grazie e arrivederci NON CI INTERESSA!!*"[1]

The first example is normal because it does not contain hate, insults, intimidation, violence or discrimination.

2. "*...Sta settimanale passerella dello #sciacallo #no #proprioNo! Ascoltare un #pagliaccio padano dopo un vero PATRIOTA un medico di #Bergamo non si può reggere ne vedere ne ascoltare. Giletti dovrebbe smetterla di invitare certi CAZZARIPADANI! #COVID-19 #NonelArena*"[2]

The second example contains hate speech, including insults like #clown and #jackal.

3. "*Dico la mia... #Draghi è un grande economista ma a noi non serve un economista stile #Monti... A noi non serve un altro #governo tecnico per ubbidire alla lobby delle banche! A noi serve un leader politico! A noi serve un #ItalExit! A noi serve la #Lira! #No a #DraghiPremier*"[3]

The last example is a normal case, despite the strong negative sentiment. It might be controversial for the presence of the term *lobby*, often used in abusive contexts, but in this case, it is not directed against people on the basis of their race, ethnic origin, religion, gender, age, physical condition, disability, sexual orientation or political conviction.

The Inter-Annotator Agreement is k=0.53. Although this score is not high, it is in line with the score reported in the literature for hate speech against immigrants (k=0.54) (Poletto et al., 2017) and indicates that the detection of hate speech is a hard task for humans.

All the examples in disagreement were discussed and an agreement was reached between the annotators, with the help of a third supervisor. The cases of disagreements occurred more often when the sentiment of the tweet was negative, this was mainly due to:

- The use of vulgar expressions not explicitly directed against specific people but generically against political choices.
- The negative interpretation of hyper-partisan hashtags, such as #contedimettiti (#ConteResign) or #noicontrosalvini (#WeareagainstSalvini), in tweets without explicit insults or abusive language.
- The substitution of explicit insults with derogatory words, such as the word "circus" instead of "clowns".

The amount of hate labels in the original Policycorpus was 11% (1124 normal and 140 hate tweets), strongly unbalanced like the Italian HS corpus (17% of hate tweets), because it reflects the raw distribution of hate tweets in Twitter. The HaSpeeDe-tw corpus (32% of hate tweets) instead has a distribution that oversamples hate tweets and it is better for training hate speech models. Following the HaSpeeDe-tw example, in Policycorpus XL we collected more tweets of hate, randomly discarding normal tweets to reach at least 40% of hate tweets in the corpus. In the end we have 40.6% of hate labels and 59.4% of normal labels, distributed between training and test set as shown in figure 2.

We note in the style of these tweets that there is a substantial overlap among the top unigrams in the two classes, as shown in Figure 3. We suggest that weak signals, like less frequent words, are key features for the classification task.

In the next section, we report and discuss the results of classification experiments.

[1] a clear #NO to the #Netherlands that would like us to be users of the #MES economic resources but in exchange for Italy's renunciation of its budgetary autonomy. To Netherlands we say: thank you and goodbye, WE ARE NOT INTERESTED !!

[2] ... There is a weekly catwalk of the #jackal #no #notAtAll! Listening to a Padanian #clown after a true PATRIOT a doctor from #Bergamo cannot be held, seen or heard. Giletti should stop inviting certain SLACKERS FROM THE PO VALLEY! #COVID-19 #NonelArena

[3] I have my say ... #Draghi is a great economist but we don't need a #Monti-style economist ... We don't need another technical #government to obey the banking lobby! We need a political leader! We need a #ItalExit! We need the #Lira! #No to #DraghiPremier

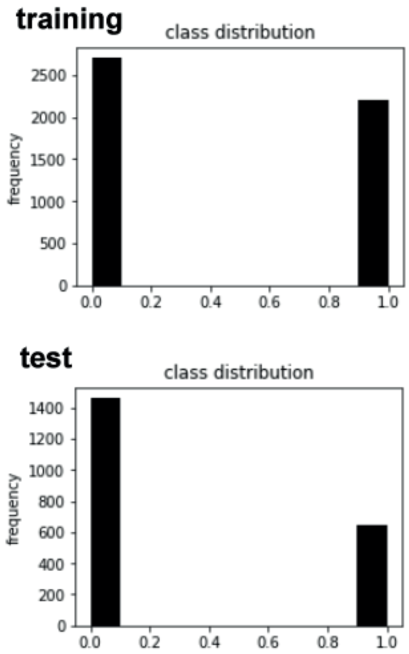


Figure 2: Distribution of classes in Policycorpus-XL training and test sets.

Figure 3: Wordclouds of the unigrams most associated to the normal and hate classes respectively. It shows a substantial overlap among the top unigrams in the two classes.

4 Baselines

In order to set the baselines for the hate speech classification task on Policycorpus-XL, we tested different classification algorithms. We are using a 70 train and 30 test percentage split, the training set shape is 4900 instances and 300 features, while the test set shape is 2100 instances and 300 features. The 300 features are the normalized frequencies of the 300 most frequent words extracted from tweets without removing the stopwords. Table 1 reports the result of classification.

algorithm	**balanced acc**	**macro F1**
majority baseline	0.500	0.37
naive bayes	0.783	0.78
decision trees	0.763	0.76
SVMs	0.788	0.79

Table 1: Results of classification with different algorithms.

We used Scikit-Learn to compute a majority baseline with a dummy classifier, that assigns all the instances to the most frequent class (normal tweets), a naive bayes classifier, a decision tree and Support Vector Machines (SVMs). The best performance for the classification of hate speech has been achieved with the SVM classifier, that has a very high precision (0.94) and poor recall (0.60). All the algorithms a The results are in line with the scores obtained by the systems on the HaSpeeDe-tw 2020 dataset at EVALITA, and we believe that there is still great room for improvement with the Policycorpus-XL, as we exploited very simple and limited features.

5 Conclusion and Future Work

We presented a large corpus of Twitter data in Italian, manually annotated with hate speech labels. The corpus is an extension of a previous one, the first corpus annotated with hate speech in the political domain in Italian.

Given the rising amount of hate messages online, not just against minorities but more and more against policies and policymakers, it is urgent to understand the phenomenon and train classifiers that could prevent people to disseminate hate in the public debate. This is very important to keep democracies alive and grant a free speech that is respectful of other people's freedom.

We plan to distribute the corpus in the next edition of EVALITA for a specific HaSpeeDe-tw task.

Acknowledgments

The research leading to the results presented in this paper has received funding from the PolicyCLOUD project, supported by the European Union's Horizon 2020 research and innovation programme under Grant Agreement no 870675.

References

Rodrigo Agerri, Roberto Centeno, María Espinosa, Joseba Fernandez de Landa, and Alvaro Rodrigo. 2021. VaxxStance@IberLEF 2021: Going Beyond Text in Crosslingual Stance Detection. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021)*. CEUR-WS.org.

Ahmet Aker, Fabio Celli, Adam Funk, Emina Kurtic, Mark Hepple, and Rob Gaizauskas. 2016. Sheffield-trento system for sentiment and argument structure enhanced comment-to-article linking in the online news domain.

Firoj Alam, Fabio Celli, Evgeny Stepanov, Arindam Ghosh, and Giuseppe Riccardi. 2016. The social mood of news: self-reported annotations to design automatic mood detection systems. In *Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES)*, pages 143–152.

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 759–760.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Nozza Debora, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, Manuela Sanguinetti, et al. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *13th International Workshop on Semantic Evaluation*, pages 54–63. Association for Computational Linguistics.

Cristina Bosco, Felice Dell'Orletta, Fabio Poletto, Manuela Sanguinetti, and Maurizio Tesconi. 2018. Overview of the evalita 2018 hate speech detection task. In *EVALITA 2018-Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian*, volume 2263, pages 1–9. CEUR.

Fabio Celli and Luca Polonio. 2015. Facebook and the real world: Correlations between online and offline conversations. *CLiC it*, page 82.

Fabio Celli, Giuseppe Riccardi, and Arindam Ghosh. 2014. Corea: Italian news corpus with emotions and agreement. In *Proceedings of CLIC-it 2014*, pages 98–102.

Fabio Celli, Evgeny A Stepanov, Massimo Poesio, and Giuseppe Riccardi. 2016. Predicting brexit: Classifying agreement is better than sentiment and pollsters. In *PEOPLES@ COLING*, pages 110–118.

Alessandra Teresa Cignarella, Mirko Lai, Cristina Bosco, Viviana Patti, and Paolo Rosso. 2020. Sardistance@evalita2020: Overview of the task on stance detection in italian tweets. In *Proceedings of the 7th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2020)*, volume 2765 of *CEUR Workshop Proceedings*, Aachen, Germany, December. CEUR-WS.org.

Fabio Del Vigna, Andrea Cimino, Felice Dell'Orletta, Marinella Petrocchi, and Maurizio Tesconi. 2017. Hate me, hate me not: Hate speech detection on facebook. In *Proceedings of the First Italian Conference on Cybersecurity (ITASEC17)*, pages 86–95.

Armend Duzha, Cristiano Casadei, Michael Tosi, and Fabio Celli. 2021. Hate versus politics: detection of hate against policy makers in italian tweets. *SN Social Sciences*, 1(9):1–15.

Karmen Erjavec and Melita Poler Kovačič. 2012. "you don't understand, this is a new war!" analysis of hate speech in news web sites' comments. *Mass Communication and Society*, 15(6):899–920.

Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)*, 51(4):1–30.

Antigoni Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. In *Twelfth International AAAI Conference on Web and Social Media*.

Fabio Giglietto, Nicola Righetti, Giada Marino, and Luca Rossi. 2019. Multi-party media partisanship attention score. estimating partisan attention of news media sources using twitter data in the lead-up to 2018 italian election. *Comunicazione politica*, 20(1):85–108.

Imane Guellil, Ahsan Adeel, Faical Azouaou, Sara Chennoufi, Hanene Maafi, and Thinhinane Hamitouche. 2020. Detecting hate speech against politicians in arabic community on social media. *International Journal of Web Information Systems*.

Rakshita Jain, Devanshi Goel, Prashant Sahu, Abhinav Kumar, and Jyoti Prakash Singh. 2021. Profiling hate speech spreaders on twitter. In *CLEF*.

Sylvia Jaki and Tom De Smedt. 2019. Right-wing german hate speech on twitter: Analysis and automatic detection. *arXiv preprint arXiv:1910.07518*.

Irene Kwok and Yuzhou Wang. 2013. Locate the hate: Detecting tweets against blacks. In *Proceedings of the twenty-seventh AAAI conference on artificial intelligence*, pages 1621–1622.

Sean MacAvaney, Hao-Ren Yao, Eugene Yang, Katina Russell, Nazli Goharian, and Ophir Frieder. 2019. Hate speech detection: Challenges and solutions. *PloS one*, 14(8):e0221152.

Stefano Menini, Giovanni Moretti, Michele Corazza, Elena Cabrio, Sara Tonelli, and Serena Villata. 2019. A system to monitor cyberbullying based on message classification and social network analysis. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 105–110.

Nuria Oliver, Bruno Lepri, Harald Sterly, Renaud Lambiotte, Sébastien Deletaille, Marco De Nadai, Emmanuel Letouzé, Albert Ali Salah, Richard Benjamins, Ciro Cattuto, et al. 2020. Mobile phone data for informing public health actions across the covid-19 pandemic life cycle.

Fabio Poletto, Marco Stranisci, Manuela Sanguinetti, Viviana Patti, and Cristina Bosco. 2017. Hate speech annotation: Analysis of an italian twitter corpus. In *4th Italian Conference on Computational Linguistics, CLiC-it 2017*, volume 2006, pages 1–6. CEUR-WS.

Fabio Poletto, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, and Viviana Patti. 2021. Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources & Evaluation*, 55:477–523.

Francisco Rangel, GLDLP Sarracén, BERTa Chulvi, Elisabetta Fersini, and Paolo Rosso. 2021. Profiling hate speech spreaders on twitter task at pan 2021. In *CLEF*.

Punyajoy Saha, Binny Mathew, Pawan Goyal, and Animesh Mukherjee. 2018. Hateminers: detecting hate speech against women. *arXiv preprint arXiv:1812.06700*.

Manuela Sanguinetti, Fabio Poletto, Cristina Bosco, Viviana Patti, and Marco Stranisci. 2018. An italian twitter corpus of hate speech against immigrants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Manuela Sanguinetti, Gloria Comandini, Elisa Di Nuovo, Simona Frenda, Marco Stranisci, Cristina Bosco, Tommaso Caselli, Viviana Patti, and Irene Russo. 2020. Overview of the evalita 2020 second hate speech detection task (haspeede 2). In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020)*, Online. CEUR.org.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678.

Kai Shu, Xinyi Zhou, Suhang Wang, Reza Zafarani, and Huan Liu. 2019. The role of user profiles for fake news detection. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 436–439.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93.

Hajime Watanabe, Mondher Bouazizi, and Tomoaki Ohtsuki. 2018. Hate speech on twitter: A pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. *IEEE access*, 6:13825–13835.

Dejin Zhao and Mary Beth Rosson. 2009. How and why people twitter: the role that micro-blogging plays in informal communication at work. In *Proceedings of the ACM 2009 international conference on Supporting group work*, pages 243–252. ACM.

Quale testo è scritto meglio? A Study on Italian Native Speakers' Perception of Writing Quality

Aldo Cerulli•, **Dominique Brunato**◇, **Felice Dell'Orletta**◇
• University of Pisa
a.cerulli1@studenti.unipi.it
◇Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC–CNR)
ItaliaNLP Lab - *www.italianlp.it*
{dominique.brunato, felice.dellorletta}@ilc.cnr.it

Abstract

This paper presents a pilot study focused on Italian native speakers' perception of writing quality. A group of native speakers expressed their preferences on 100 pairs of essays extracted from an Italian corpus of compositions written by L1 students of lower secondary school. Analysing their answers, it was possible to identify a set of linguistic features characterizing essays perceived as well written and to assess the impact of students errors on the perception of text quality. The paper describes the crowdsourcing technique to collect data as well as the linguistic analysis and results.

1 Introduction

The institution of distance learning paradigms, which has become crucial during the Covid-19 pandemic, showed the need to provide schools and universities with Natural Language Processing (NLP)-based tools to assist students, teachers and professors. Nowadays, language technologies are more and more exploited to develop educational applications, such as *Intelligent Computer-Assisted Language Learning (ICALL)* systems (Granger, 2003) and tools for automated essay scoring (Attali and Burstein, 2006) or automatic error detection and correction (Ng et al., 2013). A fundamental requirement for developing this kind of applications is the availability of electronically accessible corpora of learners' productions. Corpora created so far differ in many respects. For instance, considering the types of examined learners, they can gather productions written by L2 students or by native speakers: the former have been built for many languages (e.g. English, Arabic, German, Hungarian, Basque, Czech, Italian), while the latter are mainly available for English. In both cases, a peculiarity of existing corpora is that they are cross-sectional rather than longitudinal. A notable exception in the context of Italian as L1 – which is the focus of our contribution – is represented by *CItA (Corpus Italiano di Apprendenti L1)*, which was jointly developed by the Institute for Computational Linguistics of the Italian National Research Council (CNR) of Pisa and the Department of Social and Developmental Psychology at Sapienza University of Rome (Barbagli et al., 2016): it is the first digitalized collection of essays written by the same group of Italian L1 learners in the first two years of the lower secondary school[1].

The diachronic and longitudinal nature of *CItA* makes it particularly suitable to study the evolution of L1 writing competence over the two years, assuming that many remarkable changes in writing skills occur in this period. For instance, in their recent work, Miaschi et al. (2021) showed that it is possible to automatically learn the writing development curve of students: they extracted a wide set of linguistic features from the essays and used them to train a binary classification algorithm able to predict the chronological order of two productions written by the same pupil at different times.

The present study ranks among research based on *CItA*, but chooses a different approach from the one just mentioned: instead of tracking the development of students' writing competence, we focused on the perception of writing quality by Italian L1 speakers with the aim of understanding whether it is possible to find the linguistic features that are crucially involved in the distinction between 'better' and 'worse' essays according to our target reader.

Contributions To the best of our knowledge, this is the first paper that (i) introduces a dataset of

[1]The corpus is freely available for research goals at http://www.italianlp.it/resources/cita-corpus-italiano-di-apprendenti-l1/

evaluated essays in terms of perceived writing quality by means of a crowdsourcing task, (ii) deals with the correlation between linguistic features and perceived quality of writing and (iii) assesses the impact of students errors on quality perception.

2 Corpus Collection

As previously mentioned, the starting point of our study was the *CItA* corpus. It comprises 1,352 essays, written by 156 pupils of seven lower secondary schools in Rome (three in the historical center and four in the suburbs) during the school years 2012-2013 and 2013-2014. The productions respond to 124 writing prompts that pertain to five textual typologies: reflexive, narrative, descriptive, expository and argumentative. An additional 'common prompt' was presented at the end of each school year, in which students were asked to write a letter to advise a younger friend how to compose better essays. The common prompts were aimed at understanding how learners internalize the different writing instructions given by teachers.

Each essay contained in *CItA* is also provided by a set of metadata tracking students' biographical, sociocultural and sociolinguistic information. Beyond the longitudinal nature, the most significant novelty introduced by *CItA* regards error annotation, which was manually performed by a middle school teacher according to a new three-level schema including: the macro-class of error (i.e. grammatical, orthographic and lexical); the class of error (i.e. verbs, prepositions, monosyllabes); and the corresponding type of modification required to correct it. More details about the *CItA* collection are reported in Barbagli et al. (2016).

2.1 Essay Selection

For the purpose of our investigation, we selected 200 essays from *CItA* to be submitted to human evaluation. The essays ranged from a minimum of 141 tokens to a maximum of 1153 tokens and their average length was 359.4 tokens. Then, to gather judgments on writing quality, we created ten questionnaires, each one consisting of ten pairs of essays of the same grade, and distribute them to native speakers of all ages and cultural background.

Table 1 reports the criteria we adopted to select the pairs of essays. As it can be seen, Survey 1 allows the comparison between essays responding to the common prompts written by students attending the first or the second grades. In surveys 2-8, we

Survey	**Selection criteria**	**Number of pairs**	
		I year	II year
1	Common prompts	5	5
2	Narrative	10	0
3	Narrative	0	10
4	Reflexive	10	0
5	Reflexive	0	10
6	Descriptive	8	2
7	Expository	3	7
8	Argumentative	3	7
9	Error bins	10	0
10	Error bins	0	10

Table 1: Criteria used for pairing the essays and number of essays for each survey.

chosen essays pertaining to the same textual typology – assuming that their similarity with regard to the content could let the annotator focus on stylistic issue to orient their judgment – and paired them according to the school year in which they were written. Instead, essays in questionnaires 9 and 10 were paired according to their number of errors: for each year, we divided the range between the minimum amount of errors (0) and the maximum one (49 for the first year, 43 for the second one) into ten error bins and designed the two surveys choosing a couple of productions for each bin. Surveys comparing essays with a similar amount of errors were meant to understand which categories of errors have a greater impact on human judgment.

2.2 Human Evaluation

After designing the surveys, we moved on to their implementation using the QuestBase platform[2]. We defined a three-section structure including the filling-in instructions, the personal data entry form and the essays evaluation pages.

Filling-in instructions. The first section reported the following submission guidelines:

> Ciao!
> Il presente sondaggio è rivolto a partecipanti di madrelingua italiana. La sua compilazione richiede circa 20 minuti. Pima di proseguire, dando il consenso alla partecipazione, ti spieghiamo in cosa consiste.
> Nelle pagine che seguono leggerai dieci coppie di temi scritti da studenti del primo e del secondo anno di scuola media. I testi possono contenere un certo numero di errori. Per ciascuna coppia ti chiediamo di indicare quale dei due temi ritieni sia scritto meglio.
> Non esistono risposte giuste o sbagliate: conta semplicemente quello che pensi! Tieni presente che i temi di una stessa coppia possono trattare argomenti diversi, ma questo non deve influire sul tuo giudizio.
> La tua partecipazione al sondaggio è completamente libera. Se in qualsiasi momento dovessi cambiare idea

[2] https://story.questbase.com/

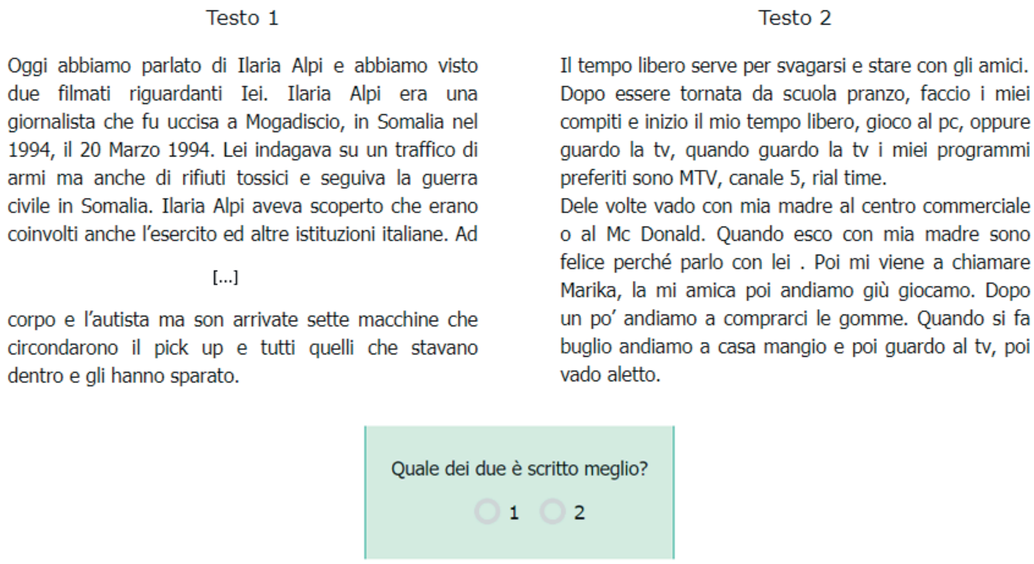


Figure 1: Comparison of a pair of essays extracted from one of the ten surveys.

> e volessi interrompere il test, potrai farlo liberamente.
> Un'ultima cosa: prima di iniziare il sondaggio, ti chiediamo di darci alcune tue informazioni anagrafiche, che serviranno solo a fini statistici. I dati rimarranno completamente anonimi e in nessun modo le risposte verranno associate alla tua persona.
> Se hai dubbi, curiosità o proposte di miglioramento, scrivimi all'indirizzo: a.cerulli1@studenti.unipi.it.
> Buona lettura!

For the sake of completeness, we also report an English translation of the same guidelines:

> Hello!
> This survey is addressed to Italian native speakers. Its submission requires about 20 minutes. By completing it, you give your consent to participation. Before going on, we explain to you what it consists of.
> In the following pages you will read ten pairs of essays written by Italian L1 learners during the first two years of lower secondary school. The essays may contain linguistic errors. For each pair, you are asked to choose the best written of the two essays.
> No answers are right or wrong: you only have to express your opinion! Bear in mind that the essays of a pair can concern different topics, but this must not affect your judgment.
> Your participation to the survey is completely free. You may withdraw from it at any time.
> Before starting the survey, we ask you to provide some personal information that will be used for statistical purposes. Data will remain completely anonymous and will not be connected to you in any way.
> If you have doubts, curiosities or improvement proposals, please write me to the address: a.cerulli1@studenti.unipi.it.
> Have a good read!

Personal data entry form. The surveys were obviously anonymous. However, as we mentioned before, we asked the annotators to entry some personal information (age, sex, education) for statistical purposes.

Essays evaluation. The third section comprised ten pages, each occupied by two side by side essays and a field to give the answer (Figure 1). The user had to choose the label '1' if they had preferred the first essay, '2' otherwise.

After carrying out a pilot study to test the adequacy of the structure as well as the completeness and clearness of the instructions, we started collecting evaluations. Using Linktree[3] we added the ten questionnaires links to a single web page and shared its link through WhatsApp, Facebook and Instagram: clicking on it, users were redirected to the page and could access every survey.

3 Analysis of Human Judgments

We collected 223 annotations distributed quite homogeneously among the ten surveys, except for the first one, submitted 28 times. It is worth to focus on the heterogeneous composition of the readers sample. Concerning sex, the large majority of answers (183 units, equal to 82.1%) were given by women, against the 38 (17%) by men; just two people preferred not to specify their gender.

Regarding age, we divided the group into six bins (Figure 2). The most frequent class (97 units) was '20-24 years', followed by '25-29 years' (64 units). This means that most readers (72.5%) ranged from 20 to 29 years of age. 35 evaluations (15.8%) were made by natives between 30 and 39 years of age. People belonging to the remaining bins contributed to the task for an overall 11.7%.

[3] https://linktr.ee/

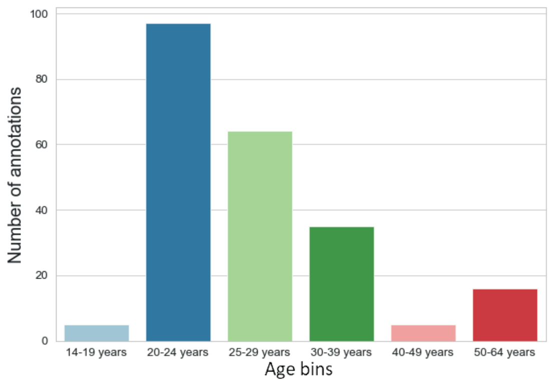


Figure 2: Distribution of annotations with respect to readers' age bins.

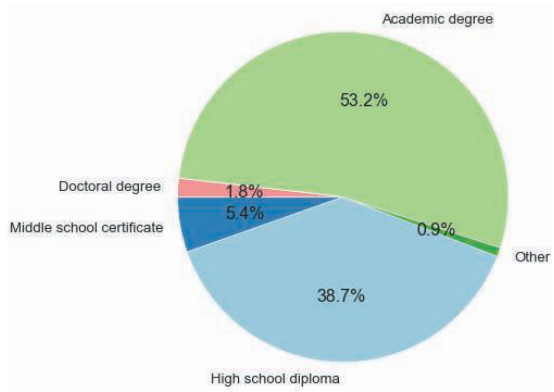


Figure 3: Distribution of annotations with respect to readers' education.

Finally, Figure 3 shows the distribution of submissions with respect to readers' education: 91.9% of annotations were given by people holding an academic degree (118 units, equal to 53.2%) or a high school diploma (86 units, equal to 38.7%). 12 annotators (5.4%) had a middle school certificate; 4 (1,8%) held a doctoral degree; the last two indicated a non-specific 'Other'.

3.1 Inter-Annotator Agreement

At this point, we defined a selection function to discard inaccurate annotations and obtain the same number of coherent annotations for each survey. Thus, we firstly built the average vector of every survey as the set of ten values '1' or '2' chosen according to the most assigned label to each pair of essays; then, we calculated the distance between each survey average vector and all its annotations. We implemented the euclidean metric generalized to the *n*-dimensional space that computes the distance between two vectors as the square root of the sum of their sizes squared difference:

$$\sqrt{\sum_{k=1}^{n}(p_k - q_k)^2} \tag{1}$$

To give relevance to the deviating degree of answers differing from the average, we assigned every pair a weight (w_k) equal to the number of times in which the 'winning' essay was chosen; then, we computed the weighted distance between annotations and average vectors.

$$\sqrt{\sum_{k=1}^{n} w_k(p_k - q_k)^2} \tag{2}$$

Finally, we ranked weighted and unweighted distance values of each survey in ascending order and calculated the Inter-annotator agreement (IAA) of the first 15 and 20 annotations. We implemented Krippendorff's alpha (α), a coefficient that expresses IAA in terms of observed (D_o) and casual (D_e) disagreement (Krippendorff, 2011):

$$\alpha = 1 - \frac{D_o}{D_e} \tag{3}$$

We noticed that IAA values of the first 15 submissions ordered by their increasing weighted distance were the highest. Thus, we took them into account (150 total annotations) for the analysis and discarded the remaining 73[4]. It is noteworthy that the selection led us to an average IAA of 0.26, that is a much higher value than the initial 0.12. Relying on the selected annotations, we established the 'winning' and 'loser' essay of each pair.

4 Data Analysis

We carried out two evaluations: a first one was meant to identify which linguistic features impact more on the human assessment of the writing quality; a second one focused on the impact of students errors on annotators' judgments. In what follows we describe the approach underlying the two perspectives and discuss our most interesting findings.

4.1 Linguistic Profiling and Stylistic Analysis

The first analysis relies on *linguistic profiling*, a NLP-based methodology in which a large set of linguistically-motivated features automatically extracted from annotated texts are used to obtain a vector-based representation of it. Such representations can be then compared across texts representative of different textual genres and varieties to identify the peculiarities of each (Montemagni, 2013;

[4]The corpus of evaluated essays is available at http://www.italianlp.it/EvaluatedEssays.zip

Feature	'Winning'		'Losers'	
	Avg.	SD	Avg.	SD
n_tokens	374.9	127.4	342.7	116.3
ttr_form_chunks_100	0.72	0.06	0.70	0.06
upos_dist_NOUN	16.31	2.49	16.98	2.63
verbs_tense_dist_Fut	2.75	4.37	2.47	6.90
verbs_form_dist_Ger	3.13	3.52	2.32	3.25
aux_mood_dist_Sub	4.41	7.22	2.48	4.51
n_prepositional_chains	10.70	6.28	9.50	5.98

Table 2: Linguistic features whose average varies significantly between the two subsets.

Feature	'Winning'		'Losers'	
	Avg.	SD	Avg.	SD
verbs_tense_dist_Fut	2.75	4.37	2.47	6.90
dep_dist_cop	1.85	0.98	1.93	1.24
dep_dist_flat:foreign	0.03	0.14	0.02	0.17
dep_dist_flat:name	0.31	0.52	0.32	0.79
dep_dist_det:predet	0.27	0.26	0.24	0.30
dep_dist_parataxis	0.13	0.21	0.15	0.31
obj_pre	31.35	13.02	30.02	15.87
verb_edges_dist_0	1.23	1.62	1.06	1.74
verb_edges_dist_1	13.45	5.44	12.48	6.30
upos_dist_CCONJ	4.17	1.28	4.51	1.61

Table 3: The 10 features that, maximally varying in 'loser' essays, are more uniform in the 'winning' ones.

van Halteren, 2004). To perform the analysis, we relied on Profiling-UD[5], a recently introduced tool that allows the extraction of a wide set of lexical, morpho-syntactic and syntactic features from texts linguistically annotated according to the Universal Dependencies (UD)[6] formalism. These features, described in details in Brunato et al. (2020), have been shown to be involved in many tasks, all related to modeling the form rather than the content of a text, such as the assessment of text readability and linguistic complexity and the identification of stylistic traits of an author or groups of authors.

We thus split our annotated corpus into two sections: one comprised all 'winning' essays and the other all 'loser' ones. Using Profiling-UD, we extracted for each text of the two subsets a feature-based vector representation. For each considered feature we calculated the average value, the standard deviation and the coefficient of variation ($\frac{SD}{Avg}$) in the two subsets and we assessed whether the variation between mean values was significant using the Wilcoxon rank sum test.

Table 2 shows the seven linguistic features whose variation turned out to be statistically significant ($p - value < 0.05$), ordered by increasing p-values. It emerges that 'winning' essays are on average longer (32.2 tokens more) than the 'losers' (*n_tokens*), a finding that may suggest that longer compositions are evaluated as more reasoned, structured and content-rich. Interestingly, this also reflects the students' perception of school writing: Barbagli et al. (2015) showed that two of the most frequent suggestions contained in essays that respond to 'common prompts' are *Leggi/scrivi molto* ("Read/write a lot") and *Lavora sodo, fai vedere che ti impegni* ("Work hard, show your dedication"). Thus, pupils possibly write more so as to show their dedication and get higher grades. Secondly, we noticed that a richer vocabulary (*ttr_form_chunks_100*) plays a crucial role in native's judgment. This is in line with another advice of the just mentioned ranking, *Usa un vocabolario ricco ed espressivo* ("Use a rich and expressive vocabulary"), that reflects teachers' encouragement to use synonyms in order to write clearer and more readable compositions. Values related to the third feature (*upos_dist_NOUN*) reveal that 'loser' essays present a slightly higher distribution of nouns. A predominant use of nouns is typical of highly informative texts (e.g. newspaper articles, laws), while genres closer to speech contain more verbs (Montemagni, 2013). Belonging to the second category, a school essay with fewer nouns is probably perceived as more coherent with its genre. Concerning verbal inflection, 'better' productions include, on average, 0.28% more future verbs (*verbs_tense_dist_Fut*), 0.81% more gerund verbs (*verbs_form_dist_Ger*) and 1.93% more subjunctive auxiliary verbs (*aux_mood_dist_Sub*). Verbal tenses differing from present and moods differing from indicative require elevated linguistic skills, which positively influence annotators' choices. The last feature significantly varying between the two groups is the number of prepositional chains (*n_prepositional_chains*): 'winning' compositions have, on average, 1.2 more of them.

A further study was focused on the variability degree of linguistic features in the two essay groups. For each subset, we ordered the features by their increasingly coefficients of variation; then, we calculated the difference between the two rankings in order to identify the features that were maximally uniformly distributed in 'better' essays as compared to the 'worse' ones (Table 3). It can be noticed that future verbs (*verbs_tense_dist_Fut*) are very uniformly distributed among 'better' essays. We

[5] http://linguistic-profiling.italianlp.it/
[6] https://universaldependencies.org/

have previously commented that their frequency is higher in the ‘winners’; it proves again that natives interpret the use of complex verbal forms as an indicator of higher skills. Also parataxis distribution (*dep_dist_parataxis*) is quite uniform in ‘winning’ essays; however, its average value is higher in the ‘loser’ ones. It can be deduced that annotators prefer hypotaxis but this is not surprising: hypotactic periods are more structured and elegant and require refined abilities to be built. The same evidence is given on the morphosyntactic level (*upos_dist_CCONJ*), since ‘better’ compositions include 0.34% less coordinating conjunctions. Curiously, ‘better’ essays have, on average, 0.1% more foreign terms (*dep_dist_flat:foreign*); this may suggest that annotators appreciate these expressions. Finally, it is worth highlighting a higher and more uniform percentage of verbs with few modifiers in the ‘winning’ essays (*verb_edges_dist_0*, *verb_edges_dist_1*).

4.2 Students Errors Impact

The last analysis was aimed at assessing whether and in what measure students errors impact on human judgments. We counted the pairs of essays whose ‘winning’ composition had a lower number of errors, those in which the ‘loser’ one had more mistakes and those with an equal number of errors. We noticed that essays with fewer errors had won in 56% cases, reaching the 79% if including pairs with the same number of errors. This procedure gave a first empirical answer to our starting question: errors substantially affect human assessment.

At this point, we focused on error categories to identify which ones affect more the perception of writing quality. For each category, we calculated the average number of errors and their standard deviation in both subsets; then, relying on Wilcoxon rank sum test, we found out that grammatical and orthographic mistakes vary significantly between the two groups (Table 4). As expected, ‘loser’ essays have, on average, 1.29 more grammatical errors and 0.85 more orthographic errors. It is worth to add that orthographic mistakes variation ($p - value = 0.007$) is more significant than the other ($p - value = 0.029$). This could mean that natives judge deviations in orthography worse than those in grammar. Once again, our findings are in line with Barbagli et al. (2015): *Usa una corretta ortografia* (“Use correct orthography”) is the 2nd of the most frequent suggestions given in the second year; moreover, *Errori di ortografia* (“Orthography errors”) occupies the 6th and the 1st position among the most salient terms respectively of the first and the second year. The non-significant variations of lexical ($p - value = 0.581$) and punctuation errors ($p - value = 0.617$) are probably due to their scarce amount in the analysed essays.

Category	**‘Winning’ essays**		**‘Loser’ essays**	
	Avg.	**SD**	**Avg.**	**SD**
Grammar	3.28	5.516	4.57	6.126
Orthography	3.18	4.517	4.03	4.826

Table 4: Error categories whose average varies significantly between the two subsets.

5 Conclusions

We presented a pilot study towards the identification of the linguistic features that are own of well written perceived essays. We collected Italian natives’ preferences on 100 pairs of essays written by L1 students, that we analysed in terms of linguistic profiling and errors distribution. Our results reveal an interesting correspondence between annotators’ judging criteria and writing instructions that L1 learners receive by teachers. Our findings could be interpreted as an indicator of the reliability of our data and, more in general, could suggest the effectiveness of crowdsourcing methods to quickly build large and reliable datasets. Considering the lack of Italian corpora of graded essays, such datasets could be valuable resources for the development of Computer-Assisted Learning Systems.

The limited size of our dataset certainly reduced the amount of results. Thus, we have to expand it (i) by collecting more annotations for the already existing surveys and (ii) by creating and distributing new surveys in order to gather judgments on new pairs of essays. Analysis on the enlarged dataset could provide more features that are own of good essays. Following the model of Miaschi et al. (2021), we could use the results to train a classifier that, given a pair of essays, recognizes the best written one.

The tool would not presume to replace teachers, but it could be a valuable teaching aid. Students could use it to get an immediate and preliminary self-assessment on their written productions so as to better understand their mistakes and hopefully avoid repeating them. Such tools can be very useful if integrated into educational processes based on distance learning paradigms, which need adequate technological infrastructures to be really efficient.

References

Yigal Attali and Jill Burstein. 2006. Automated Essay Scoring With e-rater® V. 2. *The Journal of Technology, Learning, and Assessment*, 4(3).

Alessia Barbagli, Pietro Lucisano, Felice Dell'Orletta, and Giulia Venturi. 2015. Il ruolo delle tecnologie del linguaggio nel monitoraggio dell'evoluzione delle abilità di scrittura: primi risultati. *Italian Journal of Computational Linguistics (IJCoL)*, 1(1):99–117.

Alessia Barbagli, Lucisano Pietro, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2016. Cita: an L1 Italian Learners Corpus to Study the Development of Writing Competence. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 88–95, Portorož, Slovenia. European Language Resources Association (ELRA).

Dominique Brunato, Andrea Cimino, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2020. Profiling-UD: a Tool for Linguistic Profiling of Texts. In *Proceedings of the 12th Conference of Language Resources and Evaluation (LREC 2020)*, pages 7145–7151, Marseille, France. European Language Resources Association (ELRA).

Sylviane Granger. 2003. Error-tagged learner corpora and CALL: A promising synergy. *CALICO Journal*, 20(3):465–480.

Hans van Halteren. 2004. Linguistic profiling for author recognition and verification. In *Proceedings of the Association for Computational Linguistics*, pages 200–207.

Klaus Krippendorff. 2011. Computing Krippendorff's Alpha-Reliability. Technical report, University of Pennsylvania.

Alessio Miaschi, Dominique Brunato, and Felice Dell'Orletta. 2021. A NLP-based stylometric approach for tracking the evolution of l1 written language competence. *Journal of Writing Research (JoWR)*, 13(1):71–105.

Simonetta Montemagni. 2013. Tecnologie linguistico-computazionali e monitoraggio della lingua italiana. *Studi Italiani di Linguistica Teorica e Applicata (SILTA)*, pages 145–172.

Hwee Tou Ng, Siew Mei Wu, Yuanbin Wu, Christian Hadiwinoto, and Joel Tetreault. 2013. The CoNLL-2013 Shared Task on Grammatical Error Correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–12, Sofia, Bulgaria. Association for Computational Linguistics.

On the Development of Customized Neural Machine Translation Models

Mauro Cettolo, Roldano Cattoni, Marco Turchi
Fondazione Bruno Kessler, Trento, Italy
{cettolo,cattoni,turchi}@fbk.eu

Abstract

Recent advances in neural modeling boosted performance of many machine learning applications. Training neural networks requires large amounts of clean data, which are rarely available; many methods have been designed and investigated by researchers to tackle this issue. As a partner of a project, we were asked to build translation engines for the weather forecast domain, relying on few, noisy data. Step by step, we developed neural translation models, which outperform by far Google Translate. This paper details our approach, that - we think - is paradigmatic for a broader category of applications of machine learning, and as such could be of widespread utility.

1 Introduction

The field of machine translation (MT) has experienced significant advances in recent years thanks to improvements in neural modeling. On the one hand, this represents a great opportunity for industrial MT, on the other it also poses the great challenge of collecting large amounts of clean data, needed to train neural networks. MT training data are parallel corpora, that is collections of sentence pairs where a sentence in the source language is paired with the corresponding translation in the target language. Parallel corpora are typically gathered from any available source, in most cases the web, without much guarantees about quality nor domain homogeneity.

Over the years, the scientific community has accumulated a lot of knowledge on ways to address the problem of the quantitative and qualitative inadequacy of parallel data necessary to develop translation models. Among others, deeply investigated methods are: corpus filtering (Koehn et al., 2020), data augmentation such as data selection (Moore and Lewis, 2010; Axelrod et al., 2011) and back-translation (Bertoldi and Federico, 2009; Sennrich et al., 2016), model adaptation (Luong and Manning, 2015; Chu and Wang, 2018). They should be the working tools of anyone who has to develop neural MT models for specific language pairs and domains.

This paper reports on the development of neural MT models for translating forecast bulletins from German into English and Italian, and from Italian into English and German. We were provided with in-domain parallel corpora for each language pair but not in sufficient quantity to train a neural model from scratch. Moreover, from the preliminary analysis of data, the English side resulted noisy (e.g. missing or partial translations, misaligned sentences, etc.), affecting the quality of any pair involving that language. For this very reason, we focus on one of the pairs involving English we had to cover, namely Italian-English.

An overview of the in-domain data and the description of their analysis are given in Section 2, highlighting the issues that emerged. Section 3 describes the previously listed methods together with their employment in our specific use-case. Developed neural translation models are itemized in Section 4, where their performance are compared and discussed; our best models outperform by far Google Translate and some examples will give a grasp of the actual translation quality.

We think that our approach to the specific problem we had to face is paradigmatic for a broader category of machine learning applications, and we hope that it will be useful to the whole NLP scientific community.

2 Data

We were provided with two *csv* files of weather forecast bulletins, issued by two different forecast services that from here on are identified with the acronyms BB and TT. Each row of the BB csv contains, among other things, the text of the original bulletin written in German and, possibly, its translation into Italian and/or English; in the TT csv rows, the Italian bulletin is paired with its translation into German and/or English.

2.1 Statistics

BB Bulletins were extracted from the BB csv file and paired for any possible combination of languages. Each bulletin is stored on a single line but split in a few dozen fields; the average length of each field (about 18 German words) is appropriate for MT systems, which process long sentences with difficulty. Table 1 shows statistics of the training and test sets for the it-en language pair.

site	task	set	#seg	#src w	#trg w
BB	it-en	trn-nsy	30,957	626,211	505,688
		tst-nsy	20,000	376,553	298,560
		tot	50,957	1,002,764	804,248

Table 1: Statistics of the BB it-en benchmark. The label *nsy* will be clear after reading Section 3.2.

TT Bulletins were extracted from the TT csv file and paired for each language combination. Differently than the BB case, each TT bulletin was stored on a single line without any field split; since bulletins are quite long for automatic processing (on average 30 Italian words) and are the concatenation of rather heterogeneous sentences, we decided to segment them by splitting on strong punctuation. This requires a re-alignment of source/target segments because in general they differ in number. The re-alignment was performed by means of the *hunalign* sentence aligner[1](Varga et al., 2005). Table 2 shows statistics of the training and test sets for the it-en language pair.

site	task	set	#seg	#src w	#trg w
TT	it-en	trn	5,177	78,834	73,763
		tst	1,962	30,232	28,135
		tot	7,139	109,066	101,898

Table 2: Statistics of the TT it-en benchmark.

[1]github.com/danielvarga/hunalign

2.2 Analysis and Issues

As a good practice before starting the creation of MT models, data have been inspected and analyzed looking for potential problems. Several critical issues emerged, which are described in the following paragraphs.

Non-homogeneity of data - Since data originated from two distinct weather forecast services (BB and TT), first of all it must be established whether they are linguistically similar and, if so, to what extent. For this purpose, focusing on the languages of the it-en benchmarks, we measured the perplexity of the BB and TT test sets on n-gram language models (LMs) estimated on the BB and TT training sets:[2] the closer the perplexity values of a given text on the two LMs, the greater the linguistic similarity of BB and TT training sets. Table 3 reports values of perplexity (PP) and out-of-vocabulary rates (%OOV) for all test sets vs. LMs combinations.[3]

		LM trained on			
		BB trn		TT trn	
		PP	%OOV	PP	%OOV
it	BB tst	10.8	0.22	92.0	12.07
	TT tst	42.4	0.60	10.3	0.41
en	BB tst	8.9	0.14	80.1	8.49
	TT tst	65.6	2.05	12.7	0.51

Table 3: Cross comparison of BB and TT texts.

Overall, we can notice that the PP of the two test sets significantly varies when computed on in- and out-of-domain data. The PP of any given test set is 4 (42.4 vs. 10.8) to 9 (92.0 vs. 10.3) times higher when measured on the LM estimated on the text of the other provider than on the text of the same provider. These results highlight the remarkable linguistic difference between the bulletins issued by the two forecast services.

In-domain data scarcity - Current state-of-the-art MT neural networks (Section 4.1) have dozens to hundreds million parameters that have to be estimated from data. Unfortunately, the amount of provided data does not allow an effective estimation from scratch of such a huge number of parameters, as we will empirically prove in Section 4.3.

[2]3-gram LMs with modified shift beta smoothing were estimated using the IRSTLM toolkit (Federico et al., 2008).

[3]In order to isolate the genuine PP of the text, the dictionary upperbound to compute OOV word penalty was set to 0; the OOV rates are shown for this very reason.

BB English side - BB data have a major problem on the English side. In fact, looking at csv file, we realized that many German bulletins were not translated at all into English. Moreover, in the English side there are 20% fewer words than in the corresponding German or Italian sides, a difference that is not justified by the morpho-syntactic variations between languages. In fact, it happens that entire portions of the original German bulletins are not translated into English, or that a definitely more compact form is used, as in:

de: *Der Hochdruckeinfluss hält bis auf weiteres an.*
en: *High pressure conditions.*

This critical issue affects both training and test sets, as highlighted by figures in Table 1; as such, it negatively impacts both the quality of the translation models, if trained/adapted on such noisy data, and the reliability of evaluations, if run on such distorted data. A careful corpus filtering is therefore needed, as discussed in Section 3.2.

3 Methods

3.1 MT Model Adaptation

A standard method for facing the in-domain data scarcity issue mentioned in Section 2.2 is the so-called *fine-tuning*: given a neural MT model trained on a large amount of data in one domain, its parameters are tuned by continuing the training using a small amount of data from another domain (Luong and Manning, 2015; Chu and Wang, 2018). Though effective on the new in-domain data supplied for model adaptation, fine-tuning typically suffers from performance drops on unseen data (test set), unless proper regularization techniques are adopted (Miceli Barone et al., 2017). We avoid overfitting by fine-tuning our MT models with dropout (set to 0.3) (Srivastava et al., 2014) and performing only a limited number of epochs (5) (Miceli Barone et al., 2017).

3.2 Corpus Filtering

Machine learning typically requires large sets of clean data. Since rarely large data sets are also clean, researchers devoted much effort to data cleaning, the automatic process to identify and remove errors from data. The MT community is no exception. Even, WMT - the conference on machine translation - in 2018, 2019 and 2020 editions organized a Shared Task on Parallel Corpus Filtering. Koehn et al. (2020) provide details on the task proposed in the more recent edition, on participants, their methods and results. For reference purposes, organizers set up a competitive baseline based on LASER (Language-Agnostic SEntence Representations)[4] (Schwenk and Douze, 2017) multilingual sentence embeddings. The underlying idea is to use the cosine distance between the embeddings of the source and the target sentences to measure their parallelism. In a similar way we cleaned the BB noisy benchmark, filtering with a threshold of 0.9; statistics of the resulting bi-text are given in Table 4.

site	task	set	#seg	#src w	#trg w
BB	it-en	trn-cln	1,673	37,629	40,256
		tst-cln	1,011	20,280	21,657
		tot	2,684	57,909	61,913

Table 4: Stats of the filtered BB it-en benchmark.

The filtered bi-text does not suffer anymore from the imbalance number of words but it is 20 times smaller than the original one.

3.3 Data Augmentation

Since the corpus filtering discussed in the previous section removes most of the original data, further exacerbating the problem of data scarcity, we tried to overcome this unwanted side effect by means of data augmentation methods.

3.3.1 Data Selection

A widespreadly adopted data augmentation method is *data selection*. Data selection assumes the availability of a large general domain corpus and a small in-domain corpus; in MT, the aim is to extract parallel sentences from the large bilingual corpus that are most relevant to the target domain as defined by the small corpus.

On the basis of the bilingual cross-entropy difference (Axelrod et al., 2011), we sorted the sentence pairs of the OPUS collection,[5] used as general domain large dataset, according to their relevance to the domain determined by the concatenation of the BB and TT training sets. To establish the optimal size of the selection, we trained LMs - created in the same setup described in *non-homogeneity of data* paragraph of Section 2.2 - on increasing amounts of selected data and computed the PP of BB and TT test sets, separately for each side. Figure 1 plots the curves; the straight lines on

[4] `github.com/facebookresearch/LASER`
[5] `opus.nlpl.eu`

the bottom correspond to the PP of the same test sets on LMs built on the in-domain training sets.

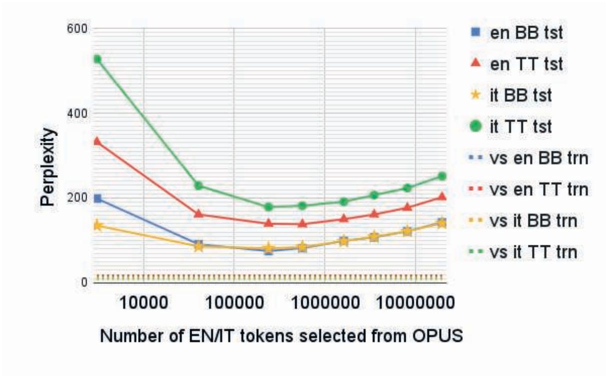


Figure 1: Perplexity of test sets on LMs estimated on increasing amounts of selected data.

The form of curves is convex, as usual in data selection. In our case, the best trade-off between the pertinence of data and its amount occur when something more than a million words is selected; therefore, we decided to mine from OPUS the bilingual text whose size is given in row *DS* of Table 5. Anyway, note that the lowest PP for selections is at least one order of magnitude greater than on LMs trained on in-domain training sets.

task	set	#seg	#src w	#trg w
it-en	DS	206,990	1,352,623	1,312,068
	BT	30,957	482,398	505,688

Table 5: Stats of selected and back translated data.

3.3.2 Back Translation

Another well-known data-augmentation method, which somehow also represents an alternative way to corpus filtering for dealing with the BB English side issue, is *back-translation*. Back-translation (Bertoldi and Federico, 2009; Sennrich et al., 2016; Edunov et al., 2018) assumes the availability of an MT system from the target language to the source language and of target monolingual data. The MT system is used to translate the target monolingual data into the source language. The result is a parallel corpus where the source side is the synthetic MT output while the target is human text. The synthetic parallel corpus is then used to train or adapt a source-to-target MT system. Although simple, this method has been shown to be very effective. We used back-translation to generate a synthetic, but hopefully cleaner, version of the BB training set. The translation into Italian of the 31k English segments of the training set (Table 1) was performed by an in-house generic en-it MT engine (details in Appendix A.1 of (Bentivogli et al., 2021)). Row *BT* of Table 5 shows the statistics of this artificial bilingual corpus; similarly to what happened with the filtering process, the numbers of Italian and English words are much more compatible than they are in the original version of the corpus.

	#segments	#src w	#trg w
it-en	32.0M	339M	352M

Table 6: Stats of the parallel generic training sets.

4 Experimental Results

4.1 MT Engine

The MT engine is built on the ModernMT framework[6] which implements the Transformer (Vaswani et al., 2017) architecture. The original generic model is *Big* sized, as defined in (Vaswani et al., 2017) by more than 200 million parameters. For training, bi-texts were downloaded from the OPUS repository[5] and then filtered through the already mentioned data selection method (Axelrod et al., 2011) using a general-domain seed. Statistics of the resulting corpus are provided in Table 6. Training was performed in the setup detailed in (Bentivogli et al., 2021).

The same Big model and its smaller variants, the *Base* with 50 million parameters and the *Tiny* with 20 million parameters, were also trained on in-domain data only for the sake of comparison.

4.2 MT Models

We empirically compared the quality of translations generated by various MT models: two generic, three genuine in-domain of different size and several variants of our generic model adapted (Section 3.1) on in-domain data resulting from the presented methods: filtering (Section 3.2), data selection (Section 3.3.1) and back-translation (Section 3.3.2). Performance was measured on the BB and TT test sets in terms of BLEU (Papineni et al., 2002), TER (Snover et al., 2006) and CHRF (Popović, 2015) scores computed by means of SacreBLEU (v1.4.14) (Post, 2018), with default

[6] `github.com/modernmt/modernmt`

MT model	BB						TT		
	noisy test set			clean test set			test set		
	%BLEU↑	%TER↓	CHRF↑	%BLEU↑	%TER↓	CHRF↑	%BLEU↑	%TER↓	CHRF↑
`Generic models:`									
GT*	11.45	106.61	.3502	32.59	51.72	.6104	32.20	61.56	.6315
FBK (Transformer_big)	7.43	113.07	.3833	19.68	63.68	.5229	23.45	70.46	.5525
`Pure in-domain models trained on` *BBtrn-nsy+TTtrn:*									
Transformer_tiny	23.34	83.86	.4882	35.80	61.05	.5808	42.19	51.79	.6488
Transformer_base	18.39	93.41	.4590	22.06	85.91	.5237	29.17	64.73	.5351
Transformer_big	20.45	95.76	.4755	24.73	89.26	.5330	28.01	68.42	.5193
`FBK model adapted on:`									
BBtrn-nsy	21.21[1]	80.82[2]	.4785[2]	37.91[3]	46.91[3]	.6172	13.77	79.14	.4007
BBtrn-cln	10.67	108.86	.4195	31.57	52.54	.5950	27.68	65.05	.5912
TTtrn	10.44	107.48	.4241	28.64	54.20	.5800	39.61	52.64	.6702
DS	10.82	109.71	.4255	30.11	54.86	.5873	29.76	63.68	.6099
BT	12.50	106.85	.4507	34.85	49.78	.6339	32.71	58.95	.6372
BBtrn-nsy+TTtrn	19.30[3]	79.29[1]	.4449	32.81	52.38	.5680	40.51[3]	51.97[3]	.6579
BBtrn-nsy+TTtrn+DS+BT	19.36[2]	86.33[3]	.4792[1]	41.17[1]	44.67[1]	.6488[2]	40.69[2]	51.84[2]	.6734[3]
BBtrn-cln+TTtrn	12.39	105.36	.4450	37.02	47.40	.6365[3]	40.34	52.16	.6755[2]
BBtrn-cln+TTtrn+DS+BT	13.75	104.59	.4619[3]	40.09[2]	45.28[2]	.6617[1]	41.16[1]	51.01[1]	.6803[1]

Table 7: BLEU/TER/CHRF scores of MT models on it-en test sets. [1], [2] and [3] indicate the "podium position" among the adapted models of each column. ($\star$) Google Translate, as it was on 14 Sep 2021.

signatures.[7]

4.3 Results and Comments

Scores are collected in Table 7. First, as expected (*in-domain data scarcity* paragraph of Section 2.2), it is not feasible to properly train a huge number of parameters with few data; in fact, the best performing pure in-domain model is the smallest one (*Transformer_tiny*). Instead, the naive application of the MT state-of-the-art would have led to simply train a *Transformer_big* model on the original in-domain data. This model would not have been competitive with *GT* on TT data (28.01 vs. 32.20 BLEU); it would have been on BB data if we had only considered the noisy test set (20.45 vs. 11.45) resulting in an important misinterpretation of the actual quality of the two systems; conversely, our preliminary analysis allowed us to discover the need of cleaning BB data, which guarantees a reliable assessment (24.73 vs. 32.59).

Data augmentation methods (*DS*, *BT*) are both effective in making available additional useful bitexts; for example, the BLEU score of the model *BBtrn-cln+TTtrn* increases by 3 absolute points (from 37.02 to 40.09) when DS and BT data are added to the adaptation corpus.

The fine-tuning of a *Transformer_big* generic model to the weather forecast domain turned out to be more effective than any training from scratch using original in-domain data only: the top performing model - *BBtrn-cln+TTtrn+DS+BT* - definitely improves the *Transformer_tiny* with respect to all metrics on the BB clean test set (40.09/45.28/.6617 vs 35.80/61.05/.5808), and to two metrics out of three on the TT test set (TER: 51.01 vs. 51.79, CHRF: .6803 vs. .6488). Moreover, all its scores are a lot better than those of Google Translate.

4.4 Examples

To give a grasp of the actual quality of automatic translations, Table 8 collects the English text generated by some of the tested MT models fed with a rather complex Italian source sentence. The manual translations observed in BB data are shown as well: their number, their variety, some questionable/wrong lexical choices in them ("high" instead of "upper-level currents", "South-western" instead of "Southwesterly") and one totally wrong ("Weak high pressure conditions.") prove the difficulty of learning from such data and the need to pay par-

[7] BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a, TER+tok.tercom-nonorm-punct-noasian-uncased, chrF2+numchars.6+space.false

`Italian source sentence:`	
	Le correnti in quota si disporranno da sudovest avvicinando masse d'aria più umida alle Alpi.
`Manual English translations found in BB bulletins:`	
	Weak high pressure conditions.
	The high currents will turn to south-west and humid air mass will reach the Alps.
	Southwesterly currents will bring humid air masses to South Tyrol.
	South-western currents will bring humid air masses to the Alps.
	South-westerly upper level flow will bring humid air masses towards our region.
	More humid air masses will reach the Alps.
	Humid air reaches the Alps with South-westerly winds.
`Automatic English translations generated by some MT models:`	
GT	The currents at high altitudes will arrange themselves from the southwest, bringing more humid air masses closer to the Alps.
FBK	Currents in altitude will be deployed from the southwest, bringing wet air masses closer to the Alps.
Transformer_tiny	South-westerly upper level flow will bring humid air masses towards the Alps.
BBtrn-cln+TTtrn+DS+BT	The upper level flow will be arranged from the southwest approaching more humid air masses to the Alps.

Table 8: Examples of manual and automatic translations.

ticular attention to the evaluation phase. Concerning translations, *GT* is able to keep most of the meaning of the source text but the translation is too literal to result in fluent English. *FBK* only partially transfers the meaning from the source and generates a rather bad English text. *Transformer_tiny* provides a very good translation both from a semantic and a syntactic point of view, losing only the negligible detail that the "air masses" are "more humid", not simply "humid". Finally, *BBtrn-cln+TTtrn+DS+BT*, the model that on the basis of our evaluations is the best one, on this specific example works very well at the semantic level but rather poorly on the grammatical level.

This example shows that pure in-domain models, as expected, are "more in-domain" than generic models, though adapted, showing greater adherence to domain-specific language. On the other hand, according to scores in Table 7, adapted models should be better in generalization. Only subjective evaluations involving meteorologists can settle the question of which model is the best.

5 Conclusions

In this paper we described the development process that led us to build competitive customized translation models. Given the provided in-domain data, we started by analyzing them under several perspectives and discovered that they are few, noisy and heterogeneous. We faced these issues by exploiting a number of methods which represent established knowledge of the scientific community: adaptation of neural models, corpus filtering and data augmentation techniques such as data selection and back-translation. In particular, corpus filtering allowed us to avoid the misleading results observed on the original noisy data, while adaptation and data augmentation proved useful in effectively taking advantage of out-of-domain resources.

References

Amittai Axelrod, Xiaodong He, and Jianfeng Gao. 2011. Domain Adaptation via Pseudo In-Domain Data Selection. In *Proc. of EMNLP*, pages 355–362, Edinburgh, Scotland, UK.

Luisa Bentivogli, Mauro Cettolo, Marco Gaido, Alina Karakanta, Alberto Martinelli, Matteo Negri, and Marco Turchi. 2021. Cascade versus Direct Speech Translation: Do the Differences Still Make a Difference? In *Proc. of ACL/IJCNLP (Volume 1: Long Papers)*, pages 2873–2887, Bangkok, Thailand.

Nicola Bertoldi and Marcello Federico. 2009. Domain Adaptation for Statistical Machine Translation with Monolingual Resources. In *Proc. of WMT*, pages 182–189, Athens, Greece.

Chenhui Chu and Rui Wang. 2018. A Survey of Domain Adaptation for Neural Machine Translation. In

Proc. of COLING, pages 1304–1319, Santa Fe, US-NM.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding Back-Translation at Scale. In *Proc. of EMNLP*, pages 489–500, Brussels, Belgium.

Marcello Federico, Nicola Bertoldi, and Mauro Cettolo. 2008. IRSTLM: An Open Source Toolkit for Handling Large Scale Language Models. In *Proc. of Interspeech*, pages 1618–1621, Brisbane, Australia.

Philipp Koehn, Vishrav Chaudhary, Ahmed El-Kishky, Naman Goyal, Peng-Jen Chen, and Francisco Guzmán. 2020. Findings of the WMT 2020 Shared Task on Parallel Corpus Filtering and Alignment. In *Proc. of WMT*, pages 726–742, Online.

Minh-Thang Luong and Christopher D. Manning. 2015. Stanford Neural Machine Translation Systems for Spoken Language Domains. In *Proc. of IWSLT*, pages 76–79, Da Nang, Vietnam.

Antonio Valerio Miceli Barone, Barry Haddow, Ulrich Germann, and Rico Sennrich. 2017. Regularization Techniques for Fine-tuning in Neural Machine Translation. In *Proc. of EMNLP*, pages 1489–1494, Copenhagen, Denmark.

Robert C. Moore and William Lewis. 2010. Intelligent Selection of Language Model Training Data. In *Proc. of ACL (Short Papers)*, pages 220–224, Uppsala, Sweden.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proc. of ACL*, pages 311–318, Philadelphia, US-PA.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proc. of WMT*, pages 392–395, Lisbon, Portugal.

Matt Post. 2018. A Call for Clarity in Reporting BLEU Scores. In *Proc. of WMT*, pages 186–191, Belgium, Brussels.

Holger Schwenk and Matthijs Douze. 2017. Learning Joint Multilingual Sentence Representations with Neural Machine Translation. In *Proc. of RepL4NLP*, pages 157–167, Vancouver, Canada.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving Neural Machine Translation Models with Monolingual Data. In *Proc. of ACL (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proc. of AMTA*, pages 223–231, Cambridge, US-MA.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Dániel Varga, Péter Halácsy, András Kornai, Nagy Viktor, Nagy Laszlo, N. László, and Tron Viktor. 2005. Parallel Corpora for Medium Density Languages. In *Proc. of RANLP*, pages 590–596, Borovets, Bulgaria.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proc. of NIPS*, pages 5998–6008, Long Beach, US-CA.

A Common Derivation for Parsing and Generation with Expectation-Based Minimalist Grammars (e-MGs)

Cristiano Chesi
NeTS – IUSS lab for NEurolinguistics, Computational Linguistics, and Theoretical Syntax, Pavia
cristiano.chesi@iusspavia.it

Abstract

Expectation-based Minimalist Grammars (e-MGs) are simplified versions of the (Conflated) Minimalist Grammars, (C)MGs, formalized by Stabler (Stabler 1997; Stabler 2011; Stabler 2013) and Phase-based Minimalist Grammars, PMGs (Chesi 2007; Chesi 2005; Stabler 2011). The crucial simplification consists of driving structure building only using lexically encoded categorial top-down expectations. The commitment on a top-down procedure (in e-MGs and PMGs, as opposed to (C)MGs, Chomsky, 1995; Stabler, 2011) allows us to define a core derivation that is the same in both parsing and generation (Momma & Phillips 2018).

1 Introduction

Minimalism (Chomsky 1995; Chomsky 2001) is an elegant transformational grammatical framework that defines structural dependencies in phrasal (i.e. hierarchical) terms simply relying on one core structure building operation, Merge, that combines lexical items and the result of other Merge operations. (1).a is the representative result of two ordered Merge operations (i.e. Merge(γ, Merge(α, β)) both taking the items α, β and γ directly from the lexicon, while (1).b relies on the so called Internal Merge (Move): the re-Merge of an item that was already merged in the structure.

(1) a. [γ [α, β]] *Merge* only
b. [β [γ [α, $\_{\beta}$]] *Merge + Move*

As result, Move connects the item at the edge of the structure (β) with a trace ($\_{\beta}$), a phonetically empty copy of the item that in a previous Merge operation combined with a hierarchically lower item (α in (1).b). In both (Conflated) Minimalist and Phase-based Minimalist Grammars ([C]MGs and PMGs respectively) Merge and Move are feature-driven operations, that is, a successful operation must be triggered by the relevant (categorial) features matching, and, once these features are used, they get deleted. Consequently, a feature pair is always responsible for each operation (unless specific features are left unerased after a successful operation, as in raising predicates and successive cyclic movement, Stabler 2011). One crucial difference between PMGs and MGs is that while MGs operate from-bottom-to-top, as indicated in (2), PMGs structure building operations apply top-down as schematized in (3)[1]:

(2) $\mathit{Merge}(\alpha_{=X}, {}_{X}\beta) = [\alpha\ [\alpha_{\cancel{=X}}\ {}_{\cancel{X}}\beta]]$ *MGs*
$\mathit{Move}({}_{+Y}\alpha, [\dots\ \beta_{-Y}\ \dots]) =$
$[\alpha\ [\beta_{\cancel{-Y}}\ [{}_{\cancel{+Y}}\alpha\ [\dots\ \beta_{-Y}\ \dots]]]]$
(3) $\mathit{Merge}(\alpha_{=X}, {}_{X}\beta) = [\alpha_{\cancel{=X}}\ [{}_{\cancel{X}}\beta]]$ *PMGs*
$\mathit{Move}([\alpha_{=S\ \cancel{+Y}}[{}_{\cancel{Y}\ Z}\beta]]) =$
$[\alpha_{\cancel{=S}\ \cancel{+Y}}[{}_{\cancel{Y}\ Z}\ \beta]\ {}_{S}[\dots\ ({}_{=Z}\ [{}_{Z}\ \beta])\ \dots]]$

Another relevant difference between the two approaches is related to the implementation of Move: MGs use the "+/-" feature distinction and the same deletion procedure after matching, while PMGs do not use "-" features and simply assume that both "+" and "=" select categorial features, which are deleted after Merge. In PMGs, "+" features force memory storage and hence the movement (downward) of the licensed item, until the relevant prominent category identifying the moved item (*Z* in (3)) is selected. If no proper selection is found, the sentence is ungrammatical. CMG as well dispenses the grammar with the +/- feature distinction and only relies on select features (=*X*), but it must assume that feature deletion can be procrastinated (again, for instance, in

[1] *α* and *β* are lexical items, =*X* indicates the selection of *X*, where X is a categorial feature. Lexical items are tuples consisting of selections/expectations (=*X*) and categories (X, i.e. selected/expected features); for convenience, select features are expressed by rightward subscripts, and categories as leftward subscripts. Similarly, Move is driven by licensing (-*Y*, leftward subscripts) and licensors (+*Y*, rightward subscripts) features (Stabler 2011).

raising predicates). Despite the fact that, from a generative point of view, all these formalisms are equivalent and they all fall under the so called mildly-context sensitive domain (Stabler 2011), it is worth to appreciate the dynamics of structure building "on-line", namely how the derivation unrolls, word by word. Taking the MGs lexicon (4), the expected constituents in (1) are built adding items to the left-edge of the structure at each Merge/Move application, as described in (5).

(4) $Lex_{MG} = \{[{}_{\mathrm{Y}}\alpha_{=\mathrm{X}}], [{}_{\mathrm{X}\,-\mathrm{Z}}\beta], [\gamma_{=\mathrm{Y}\,+\mathrm{Z}}]\}$

(5) i. $Merge({}_{\mathrm{Y}}\alpha_{=\mathrm{X}}, {}_{\mathrm{X}\,-\mathrm{Z}}\beta) = [{}_{\mathrm{Y}}\alpha\ [\alpha_{\cancel{=\mathrm{X}}}\ {}_{\cancel{\mathrm{X}}\,-\mathrm{Z}}\beta]]$

ii. $Merge(\gamma_{=\mathrm{Y}\,+\mathrm{Z}}, [{}_{\mathrm{Y}}\alpha\ [\alpha\ {}_{-\mathrm{Z}}\beta]]) =$
$[\gamma_{\cancel{=\mathrm{Y}}\,+\mathrm{Z}}\ [{}_{\cancel{\mathrm{Y}}}\alpha\ [\alpha\ \ {}_{-\mathrm{Z}}\beta]]]$

iii. $Move\ ([\gamma_{+\mathrm{Z}}\ [\alpha\ [\alpha\ {}_{-\mathrm{Z}}\beta]]]) =$
$[[{}_{\cancel{-\mathrm{Z}}}\beta]\ \gamma_{\cancel{+\mathrm{Z}}}\ [\gamma\ [\alpha\ [\alpha\ {}_{\_\beta}]]]$

An equivalent structure is obtained in PMGs[2] as shown in (7). Notice a minimal difference in the lexicon (6): the absence of the "-" features.

(6) $Lex_{PMG} = \{\ [{}_{\mathrm{Y}}\alpha_{=\mathrm{X}}], [{}_{\mathrm{Z}\,\mathrm{X}}\,\beta], [\gamma_{+\mathrm{Z}\,=\mathrm{Y}}]\ \}$

(7) i. $Merge({}_{\mathrm{Z}\,\mathrm{X}}\beta, \gamma_{+\mathrm{Z}\,=\mathrm{Y}}) = [[{}_{\cancel{\mathrm{Z}}\,\mathrm{X}}\beta]\ \gamma_{\cancel{+\mathrm{Z}}\,=\mathrm{Y}}]$ $\quad {}_{\mathrm{X}}\beta \rightarrow M$

ii. $Merge([[\beta]\ \gamma\ {}_{=\mathrm{Y}}], {}_{\mathrm{Y}}\alpha\ {}_{=\mathrm{X}}) =$
$[[\beta]\ \gamma\ [{}_{(\gamma)}\ {}_{\cancel{=\mathrm{Y}}}\ [{}_{\cancel{\mathrm{Y}}}\alpha\ {}_{=\mathrm{X}}]]$ $\quad M = \{{}_{\mathrm{X}}\beta\}$

iii. $Move([[\beta]\ \gamma\ [{}_{(\gamma)}\ [\alpha\ {}_{=\mathrm{X}}]], {}_{\mathrm{X}}\beta) =$
$[[\beta]\ \gamma\ [{}_{(\gamma)}\ [\alpha_{\cancel{=\mathrm{X}}}\ [{}_{(\alpha)}\ {}_{\cancel{\mathrm{X}}\,\_\beta}]]]]$ $\quad {}_{\mathrm{X}}\beta \leftarrow M$

The result of the two derivations is (strongly) equivalent in hierarchical (and dependency) terms. The simplicity, in pre-theoretical terms, of the two descriptions is comparable: while PMGs must postulate the *M* storage to implement Move (as result of the missing selection of a categorial feature), MGs must postulate independent workspace to build nontrivial left-branching structures, for instance before merging a multi-word subject like "the boy" with its predicate (e.g., "runs"). Furthermore, both formalisms must restrict the behavior either of the *M* buffer operativity or the accessibility to the *-f* features to limits the Move operation (e.g., island constraints, Huang, 1982).

1.1 Top-Down is Better

There are at least three reasons to commit ourselves to the top-down orientation instead of remaining agnostic or relying on the mainstream Minimalist brick-over-brick (from-bottom-to-top) approach (Chesi 2007): First, the order in which the structure is built is grossly transparent with respect to the order in which the words are processed in real-life tasks, both in generation and in parsing in PMGs, but not in MGs.

Second, in PMGs, the simple processing order of multiple expectations is sufficient to distinguish between sequential (the last expectation of a given lexical item) and nested expectations (any other expectation): The first qualifies as the transparent branch of the tree (i.e. it is able to license pending items from the superordinate selecting item), while constituents licensed by nested expectations qualify as configurational islands (Bianchi & Chesi 2006; Chesi 2015). Moreover, successive cyclic movement is easily described in PMGs without relying on feature checking at any step or non-deterministic assumptions on features deletion (Chesi 2015) contrary to (C)MGs.

A third logical reason to prefer the top-down orientation over the bottom-up alternative is related to the unicity of the root node in tree graphs. As anticipated, the creation of complex (binary) branching structures poses a puzzle for (C)MGs: Independent workspaces must be postulated, namely [the boy] and [sings …] phrases must be created before one can merge with the other:

(8) $[_{\mathrm{VP}}$ **$[_{\mathrm{DP}}$** the boy**]** **$[_{\mathrm{V}}$** sings $[_{\mathrm{DP}}$ a song]**]**]

This is the case of "complex" subject or adjunct (i.e., non-projecting constituents which are simply composed by more than one lexical item) that must be the result of (at least) one independent Merge operation, before this can merge with the relevant predicate (e.g. $[_{\mathrm{V}}$ sings …][3] in (8)). Processing these constituents represents a major difference between (bottom-up) MGs and (top-down) PMGs derivations. While MGs must decide where to start from (and both solutions are possible and forcefully logically independent from parsing or generation, which undeniably proceed "left-right"), PMGs take advantage of the "single root condition" (Partee, Meulen & Wall 1993: 439) and avoid this problem:

(9) In every well-formed constituent structure tree, there is exactly one node that dominates every node.

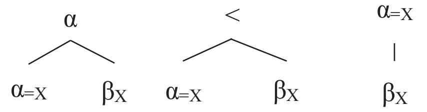
As indicated in (3), the binary operation Merge simply produces a hierarchical dependency in which the dominating (asymmetrically C-

[2] Move is implemented using a Last-In-First-Out addressable memory buffer M, where the item (*β*) with unselected categorie(s) (*X*) is stored ("${}_{X}\beta \rightarrow M$") and retrieved ("${}_{X}\beta \leftarrow M$") when selected (i.e. "=*X*").

[3] Considering the inflection "-s" as part of the lexical element or by (head) moving the root "sing-" to T is uninfluential here. This sort of head movement is implemented lexically in e-MGs (e.g. $[_{\mathrm{T}\,(=\mathrm{V}\,\mathrm{V})}$ eats $_{\ldots}]$.

commanding, in the sense of Kayne 1994) item is above the dominated (C-commanded) one. This is compatible with Stabler notation (10).a-b and plainly solves the ambiguity of the nature of the "label" of the constituent (Rizzi 2016). In this sense, PMGs (and the e-MGs discussed later) can adopt directly a more concise description, that is (10).c, more transparent with respect to the (Universal) Dependency approach (Nivre et al. 2017): Elements are "dependent" when they Merge.

(10) a. MGs b. (C)MGs c. (P/e-)MGs



The higher node (possibly the root) is always the selecting item (a *probe*, in minimalist terms), and it is the first item to be processed. This does not necessarily imply that this item is linearized before the selected category (the *goal*, in minimalist terms): if the selecting node has multiple selection needs, it must remain to the right-edge of the structure to license, locally, the other(s) selection expectation(s). E.g., if $[\alpha_{=X=Y}]$, $[_X\beta]$ and $[_Y\gamma]$, then:

(11) $[\alpha_{\cancel{=X}=Y}\ [_{\cancel{X}}\beta]\ [(\alpha_{\cancel{=Y}})\ [_{\cancel{Y}}\gamma]]]$

In this case, <α, β, γ> would be the default linearization, but it is easy to derive <β, α, γ> instead, assuming a simple parameterization on spell-out in case of multiple select features.

Here, I will argue that we can push further this intuition and only rely on (categorial) expectations, encoded in the lexical items, to guide the derivation. This leads to the so-called expectation-based Minimalist Grammars (e-MGs).

In the following sections, I will sketch a simple formalization for e-MGs (§2), and the core derivation algorithm (§3) that would be used both in Generation and Parsing tasks (§3.2).

2 The Grammar

As (C/P)MGs, e-MGs include a specification of a lexicon (*Lex*) and a set of functions (*F*), the structure building operations. The lexicon, in turn, is a finite set composed by words each consisting of phonetic/orthographic information (*Phon*) and a combination of categorical features (*Cat*), expressing *expect(ations)*, *expected* and *agreement* categories[4]. In the end, an optional set of Parameters (*P*) (see Chesi 2021), inducing minimal modifications to the structure building operations *F* and, possibly, to the *Cat* set, under the fair assumption that *F* and *Cat* are universal. More precisely, any e-MG is a 5-tuple such that:

(12) *G* = (*Phon*, *Cat*, *Lex*, *F*, *P*), where

Phon, a finite set of phonetic/orthographic features (i.e., orthographic forms representing words, e.g., "the", "smiles")
Cat, a finite set (morphosyntactic categories, that can be *expect*, *expected* or *agreement* features e.g., "D", "V"… "gen(der)", "num(ber)", "pl(ural)" etc.)
Lex, a set of expressions built from *Phon* and *Cat* (the lexicon)
F, a set of partial functions from tuples of expressions to expressions (the structure building operations)
P, a finite set of minimal transformations of *F* and/or *Cat* (the parameters), producing *F'* and *Cat'*, respectively.

2.1 Lexical Items and Categories

Each lexical item *l* in *Lex*, namely each word, is a 4-tuple defined as follows[5]:

(13) l = (*Ph*, *Exp*(*ect*), *Exp*(*ect*)*ed*, *Agr*(*ee*)),

Phon, from *Phon* in *G* (e.g., "the")
Exp, a finite list of ordered features from *Cat* in *G* (the category/ies that the item expects will follow, e.g., =*N*)
Exped is a finite list of ordered features from *Cat* in *G* (the category/ies that should be licensed/expected, e.g., *N*)
Agr(ee) is a structured list of features from *Cat* in G (e.g., *gen.fem*, *num.pl*)

All *Exp(ect)*, *Exp(ect)ed* and *Agr(ee)* features are then subsets of *Cat* in G. In *Agr*, for instance, a feminine gender specification (*gen.fem*) expresses a subset relation (i.e., "feminine" ⊆ "gender").

For sake of simplicity, each *l* will be represented as $[_{\text{Expected(; Agree)}}\ \text{Phon}\ _{=/+\text{Expect}}]$ as in (14):

(14) $[_D \text{the}\ _{=N}]$, $[_{N;\ num.pl}\ \text{dogs}]$, $[_T\ \text{barks}\ _{=D}]$

[4] As in MGs, lexical items could be specified both for phonetic (*Phon*) and semantic features (*Sem*). In e-MGs, *expectations* (=/+X) and *expectees* (X) correspond to MGs *selectors/licensors* and *selectees/licensees* respectively. *Agreement* features indicate categorial values to be unified (Chesi 2021).

[5] This is the simplest possible implementation. Attribute-Value Matrices, as in HPSH (Pollard & Sag 1994) or TRIE/compact trees exploiting the sequence of expectations (Chesi 2018; Stabler 2013) are possible implementations.

We refer to the most prominent (i.e., the first) *Expected* feature as the Label (L) of the item. E.g., the label L of "the" will be D, while the label of "barks" will be T. Similarly, let us call S (for select) the first *Expect* feature and R the remaining *Expect(actions)* (if any).

2.2 Structure Building Operations

Given l_x an arbitrary item such that $l_x = (P_x, L_x/Exped_x, S_x/R_x/Exp_x, Agr_x)$ we can define MERGE as follows:

(15) MERGE($l_{1(S1)}$, $l_{2(L2)}$) =

$$\begin{cases} 1, [l_{1(\cancel{S1})}[l_{2(\cancel{L2})}]] & if\ S_1 = L_2 \\ 0 & otherwise \end{cases}$$

MERGE is implemented as the usual binary function that is successful (it returns "1") and creates the dependency (asymmetric C-command or inclusion, in set theoretic terms) (10).c, namely [l_1 [l_2]], if and only if the label of the subsequent item (l_2) is exactly the one expected by the preceding item (l_1), namely $S_1 = L_2$. This is probably both too strict in one sense (adjuncts are not properly selected) and too permissive in another (certain elements must agree to be merged). In the first case, I assume that [l_1 [l_2]] can be formed even if S_1 is not $=X$ but $+X$: while $=X$ corresponds to functional selection (in compositional semantics terms Heim & Kratzer 1998), $+X$ corresponds to an intersective compositional interpretation (e.g. adjuncts and restrictive relative clauses). As for the agreement constraint, I postulate an extra (possibly parametrized) condition on MERGE, namely the sharing (inclusion) of the relevant *Agr* features associated to some specific categories.

The auxiliary functions necessary to implement Agreement are AGREE and UNIFY and can be minimally defined as follows:

(16) AGREE($l_{1(L1)}$, $l_{2(L2)}$) =

$$\begin{cases} 1 & if\ L_1 \wedge L_2 \in P\{Agr\} \rightarrow Unify(l_1, l_2) \\ 0 & otherwise \end{cases}$$

(17) UNIFY($l_{1(agr1)}$, $l_{2(agr2)}$) =

$$\begin{cases} 1, a, \forall a{:}Agr_1 \forall b{:}Agr_2\ a \cap b & if\ a \subseteq b \\ 1, b, \forall a{:}Agr_1\ \forall b{:}Agr_2\ a \cap b & if\ b \subseteq a \\ 0 & otherwise \end{cases}$$

Unification is simply expressed as an inclusion relation returning true and the most specific feature for any possible featural intersection between l_1 and l_2 *Agr* features[6]. Notice that Agreement is a conditional, parametrized option, that is, it only involves specific categories (possibly specified in the parameter set P): if the L category belongs to the Agreement set (Agr) in P for the grammar G, unification will be attempted, otherwise agreement will be trivially successful. The fact that AGREE should apply in conjunction with MERGE is straightforward in the D-N domain: in most Romance languages, in which gender and number are shared between the determiner and the noun, we assume that D selects N (this happens also for intermediate functional specifications, according to the cartographic intuition, Cinque 2002). This is less evident in the Subject – Predicate case, in SV language, where the predicate should select (then precede) D. Since the subject is clearly processed (i.e. merged) before T, in canonical SV sentences, and it does not select T, a re-merge operation should be considered (e.g. case checking). This re-merge (inducing the locality of Agree, *pace* Chomsky 2001) is logically and empirically sound (movement and agreement can be related and parametrized, Alexiadou & Anagnostopoulou 1998). In this case, re-merge must be preceded by MOVE, an operation that stores in memory an item which is "not fully" expected (i.e. there are $exped_2$ features remaining) by the previous MERGE:

(18) MOVE($l_{1(M1)}$, $l_{2(L2)}$) =

$$\begin{cases} 1, Push(M_1, l_{2(Phon_2=\emptyset)}) & if\ L_2 \neq \emptyset) \\ 0 & otherwise \end{cases}$$

The definition of MOVE tells us that an item (l_2) must be moved (pushed[7]) into the memory buffer (M_1) of the superordinate item (l_1) if it still has expected features to be selected ($L_2 \neq \emptyset$). Notice that item moved in M_1 is not an exact copy of l_2: the used features (including *Phon*) will not be stored in memory. This definition produces the expected derivation if it applies right after MERGE, that is, once the item l_2 is properly (at least partially) selected; in this case, if l_2 still has *exp(ect)ed* features to be licensed, it must hold in the memory buffer of the selecting item, waiting for a proper selection of what has become the new l_2 label (i.e. L_2). (Re-)Merge is then when agreement will be attempted (i.e. *if* MERGE(l_1, l_2) in §3, should then be interpreted as *if* MERGE(l_1, l_2) $\wedge$ AGREE(l_1, l_2) *then…* for specific parameterized categories). In the end, the top-down derivation in SV languages would unroll as follows: the subject (a DP) is first

[6] UNIFY(*num*, *num.pl*) = *num.pl*; UNIFY(∅, *num.pl*) = *num.pl*; UNIFY(*gen.f*, *num.pl*) = *gen.f*, *num.pl*, since *gen* and *num* are distinct agree subsets. On the other hand, UNIFY([*gen.f*, *num.sg*], *num.pl*) would fail.

[7] PUSH and POP are trivial functions operating on arrays: insert (PUSH) / remove (POP) an item to/from the first available slot of a stack or a priority queue.

selected by a superordinate item (presuppositional subject position, situation topic, focus etc.)[8] then it gets (partially) stored in the *M* buffer of the selecting item in virtue of the unselected *D* features, then re-merged as soon as a proper predicate, expressing the relevant *T* category requiring agreement (*T* should be included in the parameterized *Agreement*), is merged and properly selects a *D* argument (or it selects a *V* that later selects *D*). The content of the memory buffer is transmitted (inherited) through the last selected expectation, namely when the expecting and the expectee items successfully merge and the expecting item has no more expectations ($R_1 \neq \emptyset$).

If the expecting item has expectations, then the expected item constitutes a nested expansion, and the inheritance mechanism is blocked:

(19) INHERIT($l_{1(M1)}$, $l_{2(M2)}$) =
$$\begin{Bmatrix} 1, M_2 \Leftarrow M_1 & if\ \text{MERGE}(l_1, l_2)\ \wedge R_1 \neq \emptyset) \\ 0 & otherwise \end{Bmatrix}$$

The M buffer of the last selected item that does not have other expectations (namely a right phrasal edge, i.e., $S=\emptyset$) must be empty (i.e., $M=\emptyset$). If not, the derivation fails (i.e., it stops) since a pending item remains unlicensed:

(20) SUCCESS($l_{x(Sx,\ Mx)}$) =
$$\begin{Bmatrix} 1, & if\ S_x = \emptyset\ \rightarrow M_x = \emptyset) \\ STOP & otherwise \end{Bmatrix}$$

Notice that the sequential item must be properly selected ($=S_X$). If this is not the case, the inheritance would transmit the content of the memory buffer of the superordinate phase into the memory buffer of an adjunct or a restrictive relative clause, which clearly qualify as (right-branching) islands. Therefore, the "restrictive" (since feature driven) MERGE definition in (15) seems correct and empirically more accurate than "free Merge" (Chomsky, Gallego & Ott 2019: 238).

3 The Derivation Algorithm

We can now define the full-fledged top-down derivation algorithm which is common both to generation and to parsing tasks (§3.2). Consider *cn* to be the *current node*, *exp* the list of pending expectations and *mem* the ordered list of items in memory. We initialize our procedure by picking up an arbitrary node from *G.Lex* as *cn*. Being *cn* the *root* node of our derivation(al tree) and *w* the array of words we want to produce/recognize, we can define the function DERIVE(*cn*, *w*) as follows:

```
while cn.exp & w
  while cn.mem
    foreach cn.mem[i] in cn.mem
      if MERGE(cn.exp[0], cn.mem[i])
            POP(cn.exp)
            POP(cn.mem)
      else break
  if MERGE(cn.exp[0], w[0])
    POP(cn.exp)
    if w[0].exped
      MOVE(cn, w[0])
    if w[0].exp
      cn = w[0]
      INHERIT(exp[0], w[0])
      SUCCESS(w[0])
    POP(w)
    if not cn.exp
      while !cn.exp & (cn != root)
            cn = cn.father
  else fail
```

Informally speaking, as long as we have lexical items to consume (*w*), we loop into the set of expectations of *cn (cn.exp)*, first attempting to Merge items from (*cn.*)*mem* (if any), as in the active filler strategy (Frazier & Clifton 1989), then consuming words in the input (being *w[0]* the first available word). Remember that each word has *exp(ect)ed* features (the first being the label *L*), *exp(ectations)* and *agr(eement)* features. *Cn*s have their own *mem* that can be inherited only by the last expected item, and, apart from the root node, a *father*. The derivation is then a depth-first, left-right (i.e., real-time) strategy to derive a structure given a grammar, a root node, and a sequence of lexical items to be integrated.

3.1 The Complexity of Lexical Ambiguity

Ignoring Parameters, the derivation procedure in §3 should face lexical ambiguity: the same *Phon* in *w[n]* might be associated to multiple items *l* in *Lex* with different features; the default option is to initialize a new derivational tree for any ambiguous item in *Lex*. Given an ambiguity rate *m* in *Lex*, the derivation procedure would have an exponential order of complexity $O(m^n)$. We can mitigate this, either by selecting the element(s) bringing only coherent (i.e. expected) categories (a categorial priming strategy, Ziegler et al., 2019) or to use a statistical oracle, following Stabler (2013), to

[8] We have various options to implement this selection: a specific feature (+focus, +topic, +presupposed etc.) can be added to the relevant item (but this would lead to a proliferation of lexical ambiguity, e.g. [$_{\text{D}}$ the …] vs [$_{\text{FOC D}}$ the …]) or we assume that certain superordinate items can select specific categories, without deleting them (e.g. [$_{+\text{D}}$ ε $_{\text{FOC}}$]). In this implementation, I will pursue this second, more economic, alternative.

limit (or rank) the number of possible alternatives. It is however important to stress that lexical ambiguity is the major source of complexity in this derivation: syntactic ambiguity is greatly subsumed by the lexicon, being the source of structural differences related to the set of categorial expectations processed and to the order in which lexical items are introduce in the derivation. With the strict version of MERGE defined in (15), no attachment ambiguity is allowed, since a matching selection must be readily satisfied as soon as the relevant configuration is created (but see Chesi & Brattico 2018). This is not the case if we would admit "free merge" instead of select/licensors-driven merge: in the first case, admitting that MERGE($l_{1(S1)}$, $l_{2(L2)}$) is possible also if $S_1 \neq L_2$, would produce a syntactic ambiguity which is (exponentially) proportional to the number of items merged in the structure. This is a crucial argument to prefer feature-driven Merge. Notice, moreover, that admitting that re-merge is also possible without proper licensors/selectors, would quickly lead to unbounded unstoppable recursion. This must be prevented if we want to avoid the *halting* problem. Therefore the licensors/selectors option seem to be a more logical, self-contained, solution.

3.2 Generation and Parsing

As far as Generation is concerned, the procedure described in §3 is integrally adopted and it is sufficient to produce the expected sentence with the associated, dependency-based, structural description. As long as the sequence of words *w* is concerned, once a root node is selected, it is easy to imagine a dynamic function, instead of the static ordered sequence *w*, that incrementally proposes items to be integrated, given the history of the derivation or, at least, the last expectation (a sort of structural priming, possibly enriched with semantic features if we add to the lexicon *Sem(antic)* specifications in addition to *Cat* and *Phon* ones).

Notice that the lexicon can include phonetically empty categories; this is not a problem for the generation procedure, that consumes input tokens one by one, and then considers a phonetically empty category on a par with phonetically realized ones, namely each item should be postulated as incoming token to be processed.

From this perspective, the Parsing procedure is minimally different since it must postulate a phonetically empty item, for instance in pro-drop languages, by deducting that the *w* sequence received in input is incomplete/incompatible with specific structural hypotheses. One proposal (Brattico & Chesi 2020) relies on inflectional morphology as an overt realization of unambiguous person and number features cliticized on the predicate, hence doubling the (null) subject. Otherwise, only after a relevant category is selected (with its agreement features) and unmatched by the current input, the empty item could be postulated. This non-determinism is exacerbated by the attachment/selection ambiguity: given [$l_{1\ =/+X}$ [$l_{2\ =/+X}$]], for instance, an incoming item with *X exp(ect)ed* feature that should be merged with l_2 first, according to the derivation algorithm provided in §3, could, in fact, be merged also with l_1, assuming that $l_{2\ =X}$ expectation can be satisfied with an empty item bearing *X* as *exp(ect)ed*. Similarly, an adjunct marked with *Y exp(ect)ed* category could be merged with both l_1 and l_2 in [l_1 [l_2]] in case of lexical ambiguity ([l_1], [$l_{1\ +Y}$], [l_2], [$l_{2\ +Y}$]). In this sense, the derivation procedure in §3 is insufficient as a full-fledged parsing strategy and must be integrated with disambiguation routines dealing with the possibilities just mentioned. It is however important to stress that these disambiguation strategies do not alter the general derivation procedure introduced here, which remains the lowest common denominator of Generation and Parsing in e-MGs.

4 Conclusions

The e-MGs formalization proposed here is a simple (parametrized) framework for comparing syntactic predictions directly with human parsing and generation performance evidence. This is possible since the core derivation algorithm is assumed to be the same in both tasks (token transparency, Miller & Chomsky 1963). While there is little to add to implement a full-fledged Generation procedure (see §3.2), as long as the Parsing perspective is concerned, the information asymmetry of this task with respect to Generation requires extra routines to be implemented, in addition to the basic derivation algorithm: lexical ambiguity must be resolved "on-line" and phonetically empty items must be postulated when needed. This creates an extra level of complexity which is however manageable under the same derivational perspective here presented: the core derivation is sufficiently specified to operate independently from parsing-specific disambiguation assumptions which operate monotonically with respect to MERGE, MOVE and AGREE. This is an ideal foothold for metrics that aim at comparing the predicted difficulty not only globally (De Santo, 2020; Graf et al., 2017) but also "on-line" that is, on a word by word basis (Chesi & Canal 2019; Chesi 2021).

Implementation:
https://github.com/cristianochesi/e-MGs

References

Alexiadou, Artemis & Elena Anagnostopoulou. 1998. Parametrizing AGR: Word order, V-movement and EPP-checking. *Natural Language & Linguistic Theory*. Springer 16(3). 491–539.

Bianchi, Valentina & Cristiano Chesi. 2006. Phases, left-branch islands, and computational nesting. *Proceedings of the 29th Annual Penn Linguistics Colloquium* (University of Pennsylvania Working Papers in Linguistics) 12.1. 15–28.

Brattico, Pauli & Cristiano Chesi. 2020. A top-down, parser-friendly approach to pied-piping and operator movement. *Lingua*. Elsevier 233(102760). 1–28. https://doi.org/10.1016/j.lingua.2019.102760.

Chesi, Cristiano. 2005. Phases and Complexity in Phrase Structure Building. In *Computational Linguistics in the Netherlands 2004: Selected Papers of the 15th Meeting of Computational Linguistics in the Netherlands*, 59–75. UTRECHT: LOT. http://lotos.library.uu.nl/publish/issues/4/.

Chesi, Cristiano. 2007. An introduction to Phase-based Minimalist Grammars: why move is Top-Down from Left-to-Right. In *STIL - Studies in Linguistics - Vol. 1*, vol. 1, 38–75. Siena: CISCL Press.

Chesi, Cristiano. 2015. On directionality of phrase structure building. *Journal of Psycholinguistic Research* 65–89. https://doi.org/10.1007/s10936-014-9330-6.

Chesi, Cristiano. 2018. An efficient Trie for binding (and movement). In *Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018)*, vol. 2253. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85057729135&partnerID=40&md5=3c941a7524597857a24b64d671e7239a.

Chesi, Cristiano. 2021. Expectation-based Minimalist Grammars. *arXiv:2109.13871 [cs]*. http://arxiv.org/abs/2109.13871 (2 November, 2021).

Chesi, Cristiano & PAULI JUHANI Brattico. 2018. Larger than expected: constraints on pied-piping across languages. *RGG. RIVISTA DI GRAMMATICA GENERATIVA* 2008.4. 1–38.

Chesi, Cristiano & Paolo Canal. 2019. Person Features and Lexical Restrictions in Italian Clefts. *FRONTIERS IN PSYCHOLOGY*. https://doi.org/10.3389/fpsyg.2019.02105. https://www.frontiersin.org/articles/10.3389/fpsyg.2019.02105/full.

Chomsky, Noam. 1995. *The minimalist program*. Cambridge, MA: MIT press.

Chomsky, Noam. 2001. Derivation by phase. In Michael Kenstowicz (ed.), *Ken Hale: A life in language*, 1–52. Cambridge (MA): MIT Press.

Chomsky, Noam, Ángel J Gallego & Dennis Ott. 2019. Generative grammar and the faculty of language: Insights, questions, and challenges. *Catalan Journal of Linguistics* 229–261.

Cinque, Guglielmo. 2002. *Functional Structure in DP and IP: The Cartography of Syntactic Structures, Volume 1*. Oxford University Press.

De Santo, Aniello. 2020. MG Parsing as a Model of Gradient Acceptability in Syntactic Islands. In *Proceedings of the Society for Computation in Linguistics 2020*, 59–69. New York, New York: Association for Computational Linguistics. https://www.aclweb.org/anthology/2020.scil-1.7.

Frazier, Lyn & Charles Clifton. 1989. Successive cyclicity in the grammar and the parser. *Language and Cognitive Processes* 4(2). 93–126. https://doi.org/10.1080/01690968908406359.

Graf, Thomas, James Monette & Chong Zhang. 2017. Relative clauses as a benchmark for Minimalist parsing. *Journal of Language Modelling* 5(1). https://doi.org/10.15398/jlm.v5i1.157. https://jlm.ipipan.waw.pl/index.php/JLM/article/view/157 (21 June, 2021).

Heim, Irene & Angelika Kratzer. 1998. *Semantics in generative grammar* (Blackwell Textbooks in Linguistics 13). Malden, MA: Blackwell.

Huang, C.-T. James. 1982. *Logical relations in Chinese and the theory of grammar*. Cambridge (MA): MIT.

Kayne, Richard S. 1994. *The antisymmetry of syntax* (Linguistic Inquiry Monographs 25). Cambridge, Mass: MIT Press.

Miller, George A. & Noam Chomsky. 1963. Finitary Models of Language Users. In D. Luce (ed.), *Handbook of Mathematical Psychology*, 2–419. John Wiley & Sons.

Momma, Shota & Colin Phillips. 2018. The Relationship Between Parsing and Generation. *Annual Review of Linguistics* 4(1). 233–254. https://doi.org/10.1146/annurev-linguistics-011817-045719.

Nivre, Joakim, Željko Agić, Lars Ahrenberg, Lene Antonsen, Maria Jesus Aranzabe, Masayuki Asahara, Luma Ateyah, et al. 2017. Universal Dependencies 2.1.

Partee, Barbara H., Alice ter Meulen & Robert E. Wall. 1993. *Mathematical methods in linguistics* (Studies in Linguistics and Philosophy volume 30). Corrected second printing of the first edition. Dordrecht Boston London: Kluwer Academic Publishers.

Pollard, Carl Jesse & Ivan A. Sag. 1994. *Head-driven phrase structure grammar* (Studies in Contemporary Linguistics). Stanford : Chicago: Center for the Study of Language and Information ; University of Chicago Press.

Rizzi, Luigi. 2016. Labeling, maximality and the head–phrase distinction. *The Linguistic Review*. De Gruyter Mouton 33(1). 103–127.

Stabler, Edward. 1997. Derivational minimalism. In Christian Retoré (ed.), *Logical Aspects of Computational Linguistics*, 68–95. Berlin, Heidelberg: Springer Berlin Heidelberg.

Stabler, Edward. 2011. Computational Perspectives on Minimalism. In Cedric Boeckx (ed.), *The Oxford Handbook of Linguistic Minimalism*. Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199549368.013.0027. http://oxfordhandbooks.com/view/10.1093/oxfordhb/9780199549368.001.0001/oxfordhb-9780199549368-e-027 (26 April, 2021).

Stabler, Edward. 2013. Two Models of Minimalist, Incremental Syntactic Analysis. *Topics in Cognitive Science* 5(3). 611–633. https://doi.org/10.1111/tops.12031.

Ziegler, Jayden, Giulia Bencini, Adele Goldberg & Jesse Snedeker. 2019. How abstract is syntax? Evidence from structural priming. *Cognition* 193. 104045. https://doi.org/10.1016/j.cognition.2019.104045.

"La ministro è incinta": A Twitter Account of Women's Job Titles in Italian

Alessandra Teresa Cignarella[1,2], Mirko Lai[1], Andrea Marra and Manuela Sanguinetti[3]
1. Università degli Studi di Torino, Italy
2. Universitat Politècnica de València, Spain
3. Università degli Studi di Cagliari, Italy
{alessandrateresa.cignarella|mirko.lai}@unito.it,
andrea.marra.linguistica@gmail.com, manuela.sanguinetti@unica.it

Abstract

We analyze the use of feminine forms indicating professions and roles held by women in Italian. The study is based on Twitter and collects data from 2006 to 2021. This allows us to set up both the quantitative and the qualitative study in a diachronic perspective on a time span of 15 years. We observe the distribution over time of a selection of feminine job titles (i.e., minister, mayor, rector, engineer and lawyer), compared to their masculine counterparts, distinguishing in particular the following cases: use of marked forms and use of semi-marked forms. The analysis shows that the trend of using feminine (i.e. marked) forms is generally growing through time. However, the unbalance between the actual number of women employed in some professions and the use of the correspondent feminine job title is wide.

1 Introduction

The studies on how sexes are represented in language pertain to a transdisciplinary field of research where linguistic aspects intersect with psychological and social issues (Stahlberg et al., 2007). The various types of gender representations in language, along with their asymmetries, is a matter widely studied in linguistics (Hellinger and Bußmann, 2001) as well as in social psychology (Horvath et al., 2016; Hodel et al., 2017). Some of these studies have also affected Italian (Lepschy et al., 2001; Marcato and Thüne, 2002; Mucchi-Faina, 2005; Maturi, 2020), where a renewed debate has spread in the recent past on the use of a more gender-inclusive language.[1][2]

The presence of gender biases and stereotypes has drawn much attention even in the Natural Language Processing community.[3] Research in this field mainly focuses on the study of a model's performance on data associated with a certain gender, or rather on the association between gender and certain concepts as found in language models (Sun et al., 2019).

The present work, instead, aims at giving an exploratory account of the linguistic visibility of women in Italian language, with a focus in particular on job titles. For this purpose, we analyze the use of feminine forms used for job titles and professional roles in Twitter.

Studies on corpus-based discourse analysis have already focused on gender issues with respect to job titles in Italian. They either quantitatively evaluate the mostly used gendered forms in texts when referring to female referents (Formato, 2016; Formato, 2019a; Voghera and Vena, 2016), or rather assess, by means of a survey among native speakers, the degree of acceptability of some feminine job titles (Castenetto and Ondelli, 2020).

From a theoretical point of view, such works revolve (overtly or more indirectly) around the notion of *markedness* in language, that can be intended here as the "contrast between the unmarked (general, usual, non-salient) and the marked (special, emphatic)" (Clyne et al. (2009) cited in Formato (2019b, p.50)).[4] In the present context, the "gen-

[1] Elsewhere also defined as gender-fair, gender-neutral or non-sexist language (Sczesny et al., 2016).

[2] https://www.valigiablu.it/linguaggio-inclusivo-dibattito/.

[3] See, for example, the Workshop Series on Gender Bias in NLP: https://genderbiasnlp.talp.cat/.

[4] In its most general sense, this term refers to an opposition between two - otherwise equal - linguistic elements, one of which is characterized by the presence of a mark and the other by its absence (e.g. voicing in voiced *vs* voiceless stops). However, the notion underwent a number of different interpretations and applications. For an in-depth analysis of the differ-

eral, usual, non-salient" case is represented by masculine forms when used to express a generic reference. This means that grammatical masculine nouns are perceived and used as unmarked terms (for both men and women) based on the idea that they represent how the world is, opposing marked feminine terms which are seen as new, ungrammatical and 'sounding bad'.

While sharing with the studies mentioned above the same theoretical premise, the present work addresses the issue of women visibility in Italian language relying on user-generated data retrieved from Twitter: its peculiar nature as language data source, along with the opportunity it offers to extract and filter data based on specific keywords and time spans, makes this platform particularly useful for our purposes.

More precisely, we aimed at studying the distribution over time of a selection of feminine job titles, distinguishing in particular the following cases:

- the use of *marked* forms, i.e. feminine forms referring to female professionals (e.g. *la sindaca Raggi* ('mayor$_{\text{FEM}}$ Raggi'));
- (for a restricted set of examples) the use of *semi-marked* forms (Formato, 2016), i.e. the combination of masculine forms and feminine modifiers when referring to female professionals (e.g. *la neo-ministro è incinta* ('the$_{\text{FEM}}$ new$_{\text{MASC}}$ minister$_{\text{MASC}}$ is pregnant')).

We thus provide some background knowledge on the main linguistic conventions of Italian language in the assignment of grammatical gender, also mentioning some of the well-known studies that have challenged such conventions over the years, towards a more inclusive use of feminine forms, especially for professions. We then describe how data has been collected and filtered, and show the distribution of the selected job titles in both forms and across a 15-year time span.

2 Background

Italian is a grammatical gender language[5] and provides for the mandatory classification of the noun and its respective targets in agreement (modifiers, such as the adjective or the article) according to two values: masculine and feminine. The gender value is assigned according to phonological and semantic criteria (Thornton, 2005). In assigning gender to nouns denoting human referents, there is a strong tendency to semantically match grammatical gender with the sex of the referent (e.g., *la maestra è arrivata* vs. *il maestro è arrivato* - 'the teacher arrived').

Typically, the masculine is 'overextended' in reference to mixed groups (e.g. *tutti i candidati ammessi* - 'all$_{\text{MASC}}$ admitted$_{\text{MASC}}$ candidates'$_{\text{MASC}}$) or abstract functions (e.g. *le elezioni a sindaco* - 'the mayoral$_{\text{MASC}}$ elections'), as well as in the case of individuals whose gender is not (yet) known (e.g. *assumeremo un nuovo impiegato* - 'we'll hire a new employee'$_{\text{MASC}}$). However, there are cases in which, despite the existence of the feminine form, the masculine is also preferred to refer to a woman, especially when the person holds a prestigious position (Voghera and Vena, 2016). In such a case, the assignment of grammatical gender does not follow this semantic criterion: unmarked expressions referring to a woman (Thornton, 2009, p.126) or semi-marked expressions [6] are well attested. If we consider gender not only as a morphological category, but also as a semantic category, we can understand that, in the symbolic horizon within which the preceding examples move, masculine gender is taken as a neutral (or unmarked) form.

The assumed neutrality of masculine forms has already been questioned from several points of view (Cavagnoli, 2013; Thornton, 2016; Voghera and Vena, 2016). The seminal work by Alma Sabatini (1987), and the one proposed, more than two decades later, by Cecilia Robustelli (2012), have clarified the existence and use of feminine forms already provided for by the Italian linguistic system, and allowed the formulation of recommendations and guidelines for a more inclusive gendered language.

While such reform proposals went largely unheeded (Merkel et al., 2012), more recent studies seem to reveal a slight change in linguistic habits among Italian native speakers (Castenetto and Ondelli, 2020). Hence the choice to verify, by means of an analysis of user-generated content retrieved from Twitter, if a paradigm shift can be found with respect to the use of more gender-inclusive forms.

ent perspectives with which this concept is treated, we refer to Moravcsik and Wirth (1986) and Haspelmath (2006).

[5]We refer to Stahlberg et al (2007) for the complete definition of grammatical gender, natural gender and genderless languages.

[6]https://www.repubblica.it/online/speciale/presti/presti/presti.html.

3 Data Collection

Starting from the proposals presented in the recommendations of Sabatini (1987) and Robustelli (2012), we selected a shortlist of 11 job titles with both masculine and feminine endings. The selection is based on morphological criteria, more precisely on the different categories of gender suffix pairs that can be added to the root of a noun. We thus included the following terms:

Job titles ending in *-o*$_{\text{MASC}}$ / *-a*$_{\text{FEM}}$:

- *ministro/ministra* ('minister'),
- *sindaco/sindaca* ('mayor').

Job titles ending in *-tore*$_{\text{MASC}}$ / *-trice*$_{\text{FEM}}$:

- *rettore/rettrice* ('rector').

Job titles ending in *-ere*$_{\text{MASC}}$ / *-era*$_{\text{FEM}}$:

- *ingegnere/ingegnera* ('engineer').

Job titles ending in *-o*$_{\text{MASC}}$ / *-a* or *-essa*$_{\text{FEM}}$[7]

- *avvocato/avvocata/avvocatessa* ('lawyer').

Twitter recently introduced APIs (v2) that allow to access the full history of public conversations since the first tweet was created on March 21st, 2006. Accordingly, we take advantage of Twitter's full-archive search endpoint[8] for retrieving each tweet written in Italian and containing at least one of the words listed above, from March 21st, 2006 to March 21st, 2021 aiming at depicting the scenario of their use diachronically through a span of 15 years.

3.1 Data Cleaning

A preliminary data analysis shows several noisy tweets in the dataset. Some keywords are indeed particularly affected by homonymy and polysemy. For example, the keywords *sindaco* and *sindaca* are also inflections of the verb *sindacare* ('to judge, criticize, inspect'). A particular example is also the homonymy of the word *rettore* ('rector'$_{\text{MASC}}$) with the surname of a famous Italian singer and songwriter (Donatella Rettore), and of the word *avvocata* ('lawyer'$_{\text{FEM}}$) with a homonymous district of the city of Naples, Italy.

Other keywords are also affected by the use of figurative language. Particularly relevant is the use of the keywords *ministro* and *avvocata* in a religious context. Indeed, in Christianity, priests are also called *ministri di Dio* ('ministers of the Lord'), while *avvocata nostra* ('most gracious advocate') is part of the prayer 'Hail Holy Queen'. These few examples help to catch a glimpse of the difficult task of cleaning and removing noisy tweets from this dataset automatically. Therefore, we performed a semi-automatic data cleaning by using filters tailored for each word.

The final dataset consists of around 9.7 million tweets overall; Table 1 reports the number of tweets per keyword, as resulted after the cleaning process.[9] Drawing inspiration from studies in demography, where male to female ratio is a common parameter, we report the proportion of masculine (M) and feminine (F) forms in terms of M/F RATIO, where the higher the value the greater the unbalance between the two forms at the expense of the latter.

MASC	# tweets	FEM	# tweets	M/F RATIO
ministro:	3,575,613	ministra:	290,321	12.32
sindaco:	4,005,156	sindaca:	256,334	15.62
rettore:	138,328	rettrice:	4,490	30.81
ingegnere:	291,334	ingegnera:	4,759	61.22
avvocato:	1,133,456	avvocata:	22,771	49.78
		avvocatessa:	25,190	45.00
sum:	**9,143,887**	**sum:**	**405,841**	
unique:	9,090,414	unique:	378,274	

Table 1: N° of tweets retrieved for each query word.

On the numerical front we can see that the number of tweets containing the masculine form is greatly dominant. This is especially evident in the case of the keyword pair *ingegnere/ingegnera* (M/F RATIO of 61.22) despite the fact that the ratio of male and female engineers in Italy is 5.38.[10]

On the other hand, the feminine words that seem to be used in the most balanced way with respect to their masculine counterpart are *ministra* (M/F RA-

[7] The suffix *-essa* is used as a derivative for female referents starting from the male noun (Formato, 2019a), and its possible demeaning connotation has been matter of debate (Merkel et al., 2012; Mucchi-Faina, 2005). In Sabatini's Recommendations, its use is discouraged in favor of the suffix *-a* (or *-e* for some epicene nouns).

[8] `https://developer.twitter.com/en/docs/twitter-api/tweets/search/api-reference/get-tweets-search-all`.

[9] It is worth pointing out, however, that several tweets contain two or more keywords; they are counted in the table as many times as the number of keywords they contain. For this reason the values of 'sum' are higher than 'unique'.

[10] See page 13: `https://www.cni.it/images/News/2020/Iscritti_anno_2020_LQ.pdf`.

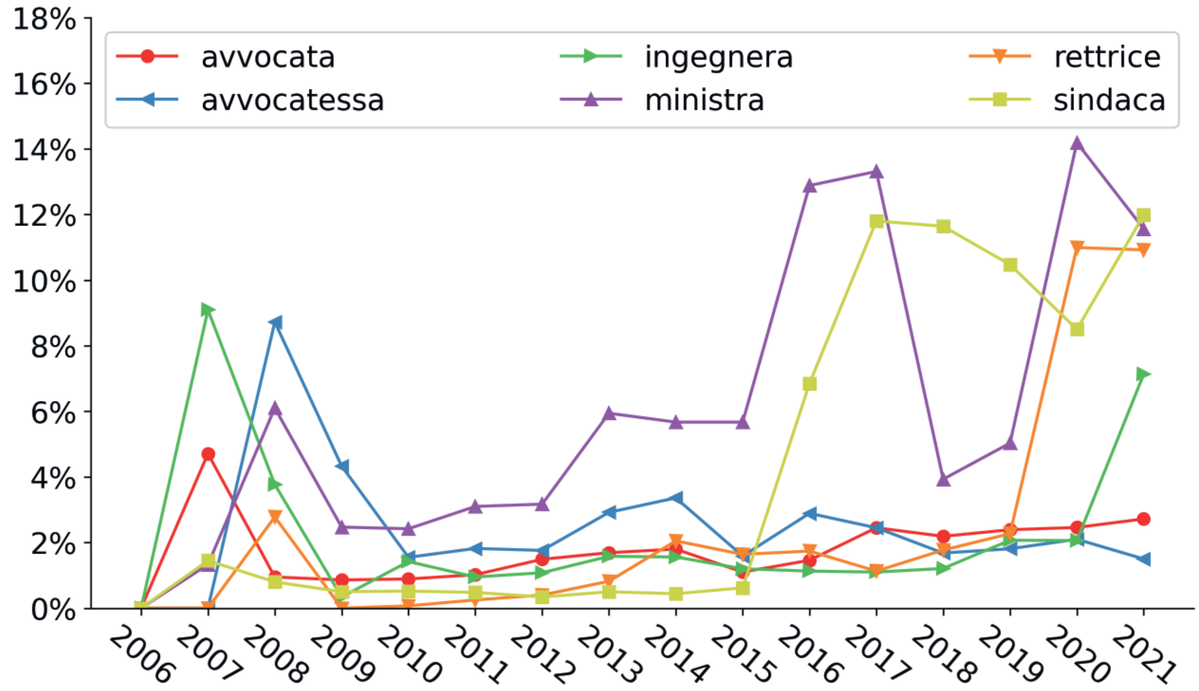


Figure 1: Frequency trend of women's job titles from 2006 to 2021.

TIO of 12.32) and *sindaca* (M/F RATIO of 15.62).

4 Data Analysis and Discussion

The first step of our data analysis consists of observing the trends of the frequency of use of the six women's job titles explored in this work.

In Figure 1 we represent the frequency of feminine job titles with respect to the total of terms used to describe the profession (FEM / FEM + MASC). We observe that from 2006 to 2021 there is a tendency to a more frequent use of female forms in general. However, relevant spikes are present on the left side of the chart. We believe they are caused by the scarcity of data before 2010, which is also imputable to the low popularity of the microblogging platform in Italy before that year. Furthermore, among the 6 sixfeminine keywords used as case study in the present work, 2 of them do not even have any occurrence in the totality of the year 2006. Their use starts with a few occurrences only from the year after (*avvocatessa* and *rettrice*).

The purple line (see Figure 1), illustrating the trend of the word *ministra* ('minister'$_{\text{FEM}}$) shows how the word has been increasingly used around 2016-2017, and then again around 2019-2020. The use of this term seem to increase during the election period and to decrease immediately afterwards. This outcome is indeed in line with the periods in which governmental changes occurred in Italy. In particular, in both those time spans there have been female ministers who have been highly politically exposed.[11]

Another fact worth mentioning is the trend of the mustard-yellow line in Figure 1 depicting the use of the word *sindaca* ('mayor'$_{\text{FEM}}$). The word seems to have started to be used more frequently in conjunction with the election of two female mayors in two large Italian cities.[12] Also the relationship between red and blue lines in the same figure presents a notable trend. Those lines respectively show the use of *avvocata* and *avvocatessa* (both: 'lawyer'$_{\text{FEM}}$). It is peculiar how the two lines show the same tendency throughout the years with the preference for the term *avvocatessa* on top of *avvocata*, until the year 2017. From that moment on, there is an inversion of trend and the occurrence of first term starts decreasing (blue line), favoring the use of the second one (red line). The oscillation between *avvocatessa* and *avvocata* therefore remains, but it seems that the latter has been increasingly gaining some ground.

The word *rettrice* ('rector'$_{\text{FEM}}$), marked by the orange line in Figure 1, has an averagely grow-

[11] Marianna Madia and Maria Elena Boschi in 2016-2017 and Luciana Lamorgese and Lucia Azzolina in 2019-2020.

[12] Virginia Raggi in Rome and Chiara Appendino in Turin.

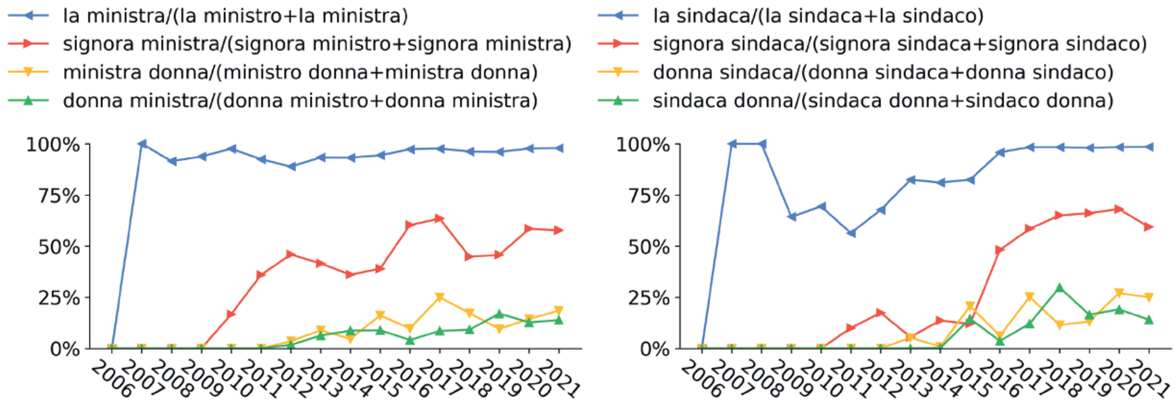


Figure 2: Ratio between the marked forms and the semi-marked forms for the terms *ministra/o* ('minister', on the left) and *avvocata/o* ('mayor', on the right).

ing distribution through time (around 2%), with a spike of increase in 2020, when – for the first time – a woman has been elected as rector in the biggest university of Europe: La Sapienza in Rome.

Finally, *ingegnera* ('engineer'$_{\text{FEM}}$) is the only one among the six terms taken into consideration with a low, though constant, trend throughout the temporal span of 15 years (around 1.6%), with only one recent spike around 2020-2021 (green line).

4.1 Analysis of N-grams

In a second step of our analysis, we aimed at investigating on the use of semi-marked forms (see Section 1). We focused on the two terms that presented the most balanced distributions with respect to their masculine counterpart (see Table 1), i.e. *ministra* and *sindaca*, and studied when and how the masculine form has been used to refer to a female referent in the real world. To do so, we extracted n-grams where one of the two tokens is one of the masculine words selected for the study and the second token is a feminine determiner or nominal modifier.

Hence, we selected the following 2-grams of interest:

- *la ministro/sindaco* ('the$_{\text{FEM}}$ minister/mayor'$_{\text{MASC}}$)
- *ministro/sindaco donna* and *donna ministro/sindaco* ('female minister/mayor'$_{\text{MASC}}$)
- *signora ministro/sindaco* ('Madame minister/mayor'$_{\text{MASC}}$)

In Figure 2 we show two charts (one for the word 'minister', and one for the word 'mayor') illustrating the ratio between the selected marked forms and the sum of such forms with semi-marked forms.

In both cases it is once again evident that the data collected before 2010 is very scarce, and that relevant statistics are, therefore, to be considered valid only after that year.

For both charts it is shown how the tendency of using marked forms (*la ministra* and *la sindaca*) is growing throughout the years; on the other hand, expressions where the female attribute is explicitly mentioned – such as *signora ministra* ('Madame minister'$_{\text{FEM}}$) and *signora sindaca* ('Madame mayor'$_{\text{FEM}}$) – are still very frequent (red lines in both charts).

Despite the outcomes derived from the analysis of n-grams, we acknowledge that the procedure described in this subsection is fairly limited. Beside the fact we studied the distribution of only two words out of the six selected for the present study, the availability of the same data enriched with part-of-speech tagging and parsing information would be highly beneficial for the automatic identification of marked and semi-marked forms.

5 Conclusion and Future Work

In this paper, we reported the results of a corpus-based account of the linguistic visibility of women in Italian language, with a focus in particular on job titles, and using Twitter as data source. From a preliminary analysis of a selection of profession nouns, we found that some marked forms are increasingly being preferred in spite of semi-marked expressions. Besides extending and systematizing this analysis to other case studies, we also aim to observe the usage of such forms by Italian native speakers by tackling the issue as a stance detection task, so to assess how the users value a given marked form and, more in general, the adoption of more gender-inclusive linguistic habits. Furthermore, the messages leveraged on this topic might overlap with the task of misogyny detection and hate speech detection as well, broadening the horizons of three different NLP detection tasks. This design choice can also be motivated with regard to *contextual stance detection* (Cignarella et al., 2020; AlDayel and Magdy, 2021), to investigate how supporters/opponents of inclusive language strategies are segregated in different online social network communities.

Finally, due to its preliminary and exploratory nature, this work only reports the distribution of feminine and masculine forms, which are the two values for gender assignment taken in consideration for the analysis. We are well aware, however, that a comprehensive study of gender-inclusive language must necessarily cover all those linguistic forms that refer to the multiple and diverse identities in the gender spectrum.

With respect to this point, innovative forms have been proposed in the last years, in order to overcome the binary opposition, even in a grammatical gender language as Italian, such as the schwa (ə), the asterisk (∗), the 'at' sign (@), and other graphic solutions. This is another aspect that is worth exploring in a stance detection perspective, so to assess users' stance regarding the use of such linguistic innovations and their spread in everyday language.

Acknowledgments

The work of A. T. Cignarella is supported by the European project 'STERHEOTYPES' funded by Compagnia di San Paolo and VolksWagen Stiftung under the 'Challenges for Europe' call. The work of M. Sanguinetti is funded by PRIN 2017 (2019-2022) project *HOPE - High quality Open data Publishing and Enrichment*.

References

Abeer AlDayel and Walid Magdy. 2021. Stance detection on social media: State of the art and trends. *Information Processing & Management*, 58(4):102597.

Giorgia Castenetto and Stefano Ondelli. 2020. The acceptability of feminine job titles in Italian newspaper articles. In Giuliana Giusti and Gabriele Iannàccaro, editors, *Language, Gender and Hate Speech. A Multidisciplinary Approach*, pages 75–90. Edizioni Ca' Foscari.

Stefania Cavagnoli. 2013. *Linguaggio giuridico e lingua di genere: una simbiosi possibile*. Edizioni dell'Orso.

Alessandra Teresa Cignarella, Mirko Lai, Cristina Bosco, Viviana Patti, Rosso Paolo, et al. 2020. SardiStance@EVALITA2020: Overview of the Task on Stance Detection in Italian Tweets. In *EVALITA 2020 Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian*. CEUR-WS.org.

Michael Clyne, Catrin Norrby, and Jane Warren. 2009. *Language and human relations: Styles of address in contemporary language*. Cambridge University Press.

Federica Formato. 2016. Linguistic markers of sexism in the Italian media: a case study of ministra and ministro. *Corpora*, 11(3):371–399.

Federica Formato. 2019a. *Gender, Discourse and Ideology in Italian*. Palgrave Studies in Language, Gender and Sexuality. Palgrave Macmillan.

Federica Formato. 2019b. An overview of grammatical gender in italian. In Federica Formato, editor, *Gender, Discourse and Ideology in Italian. Palgrave Studies in Language, Gender and Sexuality*, pages 39–80. Palgrave Macmillan.

Martin Haspelmath. 2006. Against Markedness (And What to Replace It With). *Journal of Linguistics*, 42(1):25–70.

Marlis Hellinger and Hadumod Bußmann. 2001. *Gender across languages. The linguistic representation of women and men*, volume 1 of *Impact: Studies in language and society*. John Benjamins Publishing Company.

Lea Hodel, Magdalena Formanowicz, Sabine Sczesny, Jana Valdrová, and Lisa von Stockhausen. 2017. Gender-fair language in job advertisements: A cross-linguistic and cross-cultural analysis. *Journal of Cross-Cultural Psychology*, 48(3):384–401.

Lisa K. Horvath, Elisa F. Merkel, Anne Maass, and Sabine Sczesny. 2016. Does gender-fair language pay off? The social perception of professions from a cross-linguistic perspective. *Frontiers in Psychology*, 6.

Anna Laura Lepschy, Giulio Lepschy, and Helena Sanson. 2001. Lingua italiana e femminile. *Quaderns d'Italià*, 6:9–18.

Gianna Marcato and Eva Maria Thüne. 2002. Gender and female visibility in Italian. In Marlis Hellinger and Hadumod Bußmann, editors, *Gender across languages. The linguistic representation of women and men. Volume II*, pages 187–217. John Benjamins Publishing Company.

Pietro Maturi. 2020. Qual è il tuo pronome? Riflessioni su questioni di genere nelle lingue europee. *Fuori Luogo Rivista di Sociologia del Territorio, Turismo, Tecnologia*, 8(2/2000):67–74.

Elisa Merkel, Anne Maass, and Laura Frommelt. 2012. Shielding women against status loss: The masculine form and its alternatives in the Italian language. *Journal of Language and Social Psychology*, 31(3):311–320.

Edith Moravcsik and Jessica Wirth. 1986. Markedness - an overview. In Fred R. Eckman, Edith A. Moravcsik, and Jessica R. Wirth, editors, *Markedness*, pages 1–11. Springer.

Angelica Mucchi-Faina. 2005. Visible or influential? Language reforms and gender (in)equality. *Social Science Information*, 44(1):189–215.

Cecilia Robustelli. 2012. Linee guida per l'uso del genere nel linguaggio amministrativo. Progetto Accademia della Crusca e Comune di Firenze. Available at: `https://www.rai.it/dl/docs/1354527785476guida_robustelli.pdf` (2021/09/04).

Alma Sabatini. 1987. Raccomandazioni per un uso non sessista della lingua italiana. In Alma Sabatini, editor, *Il sessismo nella lingua italiana*, pages 97–122. Presidenza del Consiglio dei Ministri e Commissione Nazionale per la Parità e le Pari Opportunità tra uomo e donna.

Sabine Sczesny, Magda Formanowicz, and Franziska Moser. 2016. Can gender-fair language reduce gender stereotyping and discrimination? *Frontiers in Psychology*, 7:25.

Dagmar Stahlberg, Friederike Braun, Lisa Irmen, and Sabine Sczesny. 2007. Representation of the sexes in language. In Klaus Fiedler, editor, *Social communication. A volume in the series Frontiers of Social Psychology*, pages 163–187. Psychology Press.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. 2019. Mitigating gender bias in natural language processing: Literature review. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1630–1640, Florence, Italy, July. Association for Computational Linguistics.

Anna Maria Thornton. 2005. *Morfologia*. Carocci.

Anna Maria Thornton. 2009. Designare le donne. In Giuliana Giusti and Susanna Regazzoni, editors, *Mi fai male...*, pages 115–133. Cafoscarina.

Anna Maria Thornton. 2016. Designare le donne: preferenze, raccomandazioni e grammatica. In Fabio Corbisiero, Pietro Maturi, and Elisabetta Ruspini, editors, *Genere e linguaggio. I segni dell'uguaglianza e della diversità*, pages 15–33. Franco Angeli.

Miriam Voghera and Debora Vena. 2016. Forma maschile, genere femminile: si presentano le donne. In Fabio Corbisiero, Pietro Maturi, and Elisabetta Ruspini, editors, *Genere e linguaggio. I segni dell'uguaglianza e della diversità*, pages 34–52. Franco Angeli.

GQA-it: Italian Question Answering on Image Scene Graphs

Danilo Croce[1], **Lucia C. Passaro**[2], **Alessandro Lenci**[3], **Roberto Basili**[1]
[1] Dept. of Enterprise Engineering, University of Rome "Tor Vergata"
[2] Dept. of Computer Science, University of Pisa
[3] Dept. of Philology, Literature and Linguistics, University of Pisa
croce@info.uniroma2.it, lucia.passaro@unipi.it,
alessandro.lenci@unipi.it, basili@info.uniroma2.it

Abstract

The recent breakthroughs in the field of deep learning have lead to state-of-the-art results in several Computer Vision and Natural Language Processing tasks such as Visual Question Answering (VQA). Nevertheless, the training requirements in cross-linguistic settings are not completely satisfying at the moment. The datasets suitable for training VQA systems for non-English languages are still not available, thus representing a significant barrier for most neural methods. This paper explores the possibility of acquiring in a semi-automatic fashion a large-scale dataset for VQA in Italian. It consists of more than 1 M question-answer pairs over 80k images, with a test set of 3,000 question-answer pairs manually validated. To the best of our knowledge, the models trained on this dataset represent the first attempt to approach VQA in Italian, with experimental results comparable with those obtained on the English original material.

1 Introduction

Multimodal information processing is crucial to deal with a wide array of human actions and real-world computer applications. Notably, when observing a real-world scene, agents – both human and virtual ones – should understand what kinds of objects it depicts and the relations occurring among them. Such understanding allows agents to reason about the scene and the context in which it appears, thus inferring additional information that can be used for different purposes.

In recent years, several Artificial Intelligence (AI) tasks have been proposed in order to challenge systems in drawing inferences from multimodal inputs bringing together both linguistic and visual contents. An important task boosting research in multimodal scenarios is represented by Visual Question Answering (Antol et al., 2015; Srivastava et al., 2020). This task consists of correctly answering natural language questions regarding an input image. This requires the integration of vision, language and commonsense knowledge to answer. In English, several benchmark datasets have been proposed to deal with visual reasoning and question answering (Antol et al., 2015; Hudson and Manning, 2019; Srivastava et al., 2020). However, despite the impressive advances obtained in this context thanks to both new available resources and models, other languages still lack large-scale datasets suitable to learn VQA models.

In this paper, we present the semi-automatic creation of GQA-it, a large-scale Italian dataset based on the balanced version of GQA (Hudson and Manning, 2019). Specifically, we obtained more than 1 million question/answer pairs in Italian over 80K images by applying Neural Machine Translation (NMT) and we manually validated $3,000$ examples to provide a valuable benchmark. Moreover, we adapted to Italian a state-of-the-art VQA neural architecture, namely LXMERT (Tan and Bansal, 2019), and we trained/evaluated it using GQA-it. The experimental evaluation in both languages shows comparable results. This result is particularly significant given the complexity of the task and the adoption of noisy, automatically translated material for training. To the best of our knowledge, this represents one of the first Italian VQA systems. GQA-it will be made available to the research community.

The rest of the paper is organized as follows. Section 2 summarizes related work. Section 3 describes the new GQA-it dataset. Section 4 presents the experimental evaluation obtained by creating a new model by using GQA-it. Conclusions and future work are drawn in Section 5.

2 Related Work

Available VQA Resources. Pioneering work in VQA has been made by Malinowski and Fritz (2014), collecting a dataset of 2,483 unique English questions about 1,449 real-world images. Then, Antol et al. (2015) introduced the task of Visual Question Answering, defined as follows: *Given an image and a natural language question about the image, the task is to provide an accurate natural language answer.* Both questions and answers are open-ended and can refer to different areas of the image. Indeed, VQA systems require a deep understanding of images and of the objects they depict, as well as reasoning abilities about available (multimodal) information. Along with proposing the new task, the authors also provided the very first large-scale VQA dataset, made of about 600k questions on about 200k images, taken from the Microsoft Objects in Context (MS COCO) dataset (Lin et al., 2014).

Afterwards, several other datasets on this topic have been created with the aim to pursue different goals (Goyal et al., 2017; Johnson et al., 2017; Zhu et al., 2016; Krishna et al., 2017). Notably, a common shortcomings of all these datasets is the presence of important real-world biases that are inherited also by neural models exploiting them for learning. Specifically, several studies report on the fact that models are driven by superficial correlations in the training data with the effect of lacking sufficient visual grounding (Agrawal et al., 2018; Goyal et al., 2017; Johnson et al., 2017).

To mitigate these aspects, the GQA dataset (Hudson and Manning, 2019) has been developed starting from Visual Genome (Krishna et al., 2017). The latter resource is valuable for several multimodal tasks, as it contains linguistically and visually more complex annotations. Specifically, images are annotated with the objects they contain and the relationships between them. In addition, Visual Genome contains a wide range of descriptions relative to specific portions of the image. Finally, the resource also comes with a visual question answering layer. However, Visual Genome is very complex from both a linguistic (ambiguity and redundancy) and visual (several regions describe the same objects) perspective, making it difficult to be easily used to train neural VQA models. This is the reason why additional normalization efforts have been performed to create a new resource, GQA (Hudson and Manning, 2019). From an annotation point of view, the resource is similar to Visual Genome, but with a lower linguistic and conceptual variability in terms of objects, relations, and descriptions. Moreover, to deal with the bias present in most of the VQA datasets, the authors created a rich *question engine* by exploiting objects, attributes and relations annotated in Visual Genome (Krishna et al., 2017) along with compositional patterns and lexical resources. In this work, we adopted the GQA dataset because, differring from the other ones, it challenges the reasoning capabilities of the models.

Neural models for VQA. The proliferation of shared tasks on this topic, led to a great technological enhancement in terms of pre-trained end-to-end models to perform visual question answering. A first benchmark is represented by the model proposed by Antol et al. (2015), which uses a CNN for visual feature extraction and a LSTM or Recurrent networks for language processing. The introduction of attention (Chen et al., 2015; Andreas et al., 2016; Yang et al., 2016) improved the results on the VQA benchmark allowing the model to focus on specific portions of the image. Subsequently, Teney et al. (2018) exploited object detection to perform VQA. The model employs R-CNN architecture and achieves good results. The introduction of Transformers and their success in NLP (Devlin et al., 2019) inspired works based on large-scale pre-training and fine-tuning studies on cross-modality. One of the first multimodal models of this generation was proposed by Tan and Bansal (2019) with the development of LXMERT, used in this work. LXMERT has been originally developed to work with GQA and embeds BERT, easily adaptable to Italian through its multilingual counterpart (Pires et al., 2019).

Multilingual approaches for VQA. More recently, new attempts have been devoted to Multilingual Visual Question Answering (Gupta et al., 2020). However, to the best of our knowledge, no gold VQA datasets is available for Italian. Therefore, this work aims to enable the training and evaluation of VQA methods in Italian, regardless of whether they are multilingual or not.

3 GQA-it: the Italian VQA Dataset

In order to build a valuable resource for Italian VQA, we considered the balanced version of GQA, in which the question distribution has been smoothed to obtain a more balanced and representative question/answer sample. In particular, we started from the benchmark split provided by Tan and Bansal (2019), namely the `train`[1] and `validation`[2] material. Moreover, the GQA test set is not publicly available. Therefore, we adopted the `test-dev`[3] subset, which represents a subset of the original test material, but it is defined to be highly representative of different linguistic and conceptual phenomena. Moreover, systems evaluated on this smaller dataset are generally in line with respect to the evaluations applied to the larger test set.

We aim to generate a large-scale dataset in which training and validation material is obtained via automatic neural machine translation and the test material is manually validated. This approach allows us to i.) create a benchmark test set in Italian and ii.) measure how sensitive the system is to the noise introduced by the machine translation. We thus applied Opus-NMT (Tiedemann and Thottingal, 2020), a Transformer-based Neural Machine translation trained on the OPUS parallel corpus, a large scale collection of texts semi-automatically aligned for several language pairs. We selected the model trained on the aligned subset of documents in the English/Italian pairs.[4] The quality of the translated questions is evaluated on a portion of the dataset. Notably, manual validation has been performed on 500 items, consisting of 250 random questions taken from the training set and 250 random questions taken from the test set. Given the characteristics of the texts contained in GQA (simple texts, no sub-sentence level) and the implementation simplicity and reproducibility, we decided to use the BLEU score for the evaluation. Overall, the performance reaches 0.82. This is impressively high, but quite in line with the BLEU obtained by the adopted translation model over the `Tatoeba.it.en` dataset (BLEU=0.72) composed of short sentences with syntactical complexity similar to the GQA dataset.[5]

The translation of answers (here expressed only with one or two tokens) is more problematic. In fact, many answers should be translated differently depending on the context or associated image, e.g., an answer "*bat*" can be translated as the animal "*pipistrello*" or the object "*mazza*". As suggested in Croce et al. (2019), in order to reduce such lexical ambiguity, we translated an answer by pairing it with the corresponding question. This way, we exploit the context sensitive nature of the adopted Transformer-based architecture: the answer "*mouse*" is thus correctly translated when paired with the question "*What's next to the keyboard?*", while generic translations, such as "*topo*", are systematically preferred when no context is made available. Unfortunately, the lexical variability of the automatically translated answers was problematic. In fact, the initial English material was characterized by $1,842$ possible answers types. After the automatic translation, this number increased to $3,306$. This is partially due to the cases in which the context does not improve the translation, e.g., the question "*What's at the top of the photo*?" is not really helpful to disambiguate the answer "*mouse*".

In other cases, multiple ways to translate the same lexical item exist, e.g., "*aircraft*" is translated both as "*aeromobile*" or "*aeroplano*". Finally, while answers involving singular and plural expressions were kept separated in the original dataset, gender is generally not marked in English, differently from Italian. Most of the times a context-sensitive translation inflected the translation in masculine and feminine. For example, "*little*" was translated in "*piccola*", "*piccolo*", "*piccole*" and "*piccoli*" depending on the items involved in the photo. To reduce this lexical variability, we applied a manual normalization to answers associated to more than two questions. We paired each original English answer with the translated ones, in order to manually normalize the translations. While this kind of manual validation is generally ineffective when dealing with machine translation, we considered that, by design, English GQA has a limited amount of polysemy, as questions, answers, and graph annotations have been automatically normalized to reduce the linguistic ambiguity (Hudson and Manning, 2019). In practice, when mentioning a "*sign*", answers (almost) always refer to objects such as a "*signboard*" more than a "*mark*" or a "*gesture*".[6] We preserved singular and plural forms. Actions, e.g.,

[1] `https://nlp.cs.unc.edu/data/lxmert_data/gqa/train.json`

[2] `https://nlp.cs.unc.edu/data/lxmert_data/gqa/valid.json`

[3] `https://nlp.cs.unc.edu/data/lxmert_data/gqa/testdev.json`

[4] `https://github.com/Helsinki-NLP/OPUS-MT-train/tree/master/models/it-en`

[5] The results of the model are available in the Github page.

[6] Only the word "*glass*" was used in both senses of "*bicchiere*" and "*vetro*", while all other words were generally characterized by only one sense.

"*skating*", "*jumping*" or "*sleeping*", were translated as the gerundive forms "*sta facendo skateboard*", "*sta saltando*" e "*sta dormendo*". Unfortunately, the noise introduced when translating adjectives makes the gender of such words problematic, so that we normalized all forms to the masculine gender. After this manual normalization, the number of possible answers across the dataset is $1,701$.

Table 1 shows the 50 most frequent answers in both the English and the Italian dataset, showing that the distribution is generally preserved across languages.

GQA
yes (17.6%) - *no* (17.6%) - *left* (5.2%) - *right* (5.1%) - *man* (1.2%) - *white* (1.2%) - *black* (1.1%) - *bottom* (0.9%) - *woman* (0.9%) - *chair* (0.9%) - *blue* (0.9%) - *top* (0.8%) - *table* (0.8%) - *brown* (0.8%) - *boy* (0.7%) - *gray* (0.6%) - *dog* (0.6%) - *green* (0.6%) - *bed* (0.6%) - *cat* (0.6%) - *girl* (0.6%) - *red* (0.5%) - *car* (0.5%) - *horse* (0.5%) - *color* (0.4%) - *bus* (0.4%) - *desk* (0.4%) - *large* (0.4%) - *orange* (0.4%) - *couch* (0.4%) - *small* (0.4%) - *yellow* (0.4%) - *shelf* (0.4%) - *elephant* (0.4%) - *people* (0.4%) - *shirt* (0.3%) - *train* (0.3%) - *wood* (0.3%) - *metal* (0.3%) - *truck* (0.3%) - *child* (0.3%) - *laptop* (0.3%) - *jacket* (0.3%) - *giraffe* (0.3%) - *player* (0.3%) - *field* (0.3%) - *cabinet* (0.3%) - *lady* (0.3%) - *guy* (0.3%) - *pink* (0.2%) -
GQA-it
sì (17.6%) - *no* (17.6%) - *sinistra* (5.2%) - *destra* (5.1%) - *uomo* (1.2%) - *bianco* (1.2%) - *nero* (1.1%) - *ragazzo* (1.0%) - *inferiore* (0.9%) - *donna* (0.9%) - *sedia* (0.9%) - *blu* (0.9%) - *in alto* (0.8%) - *marrone* (0.8%) - *tavola* (0.8%) - *auto* (0.6%) - *grigio* (0.6%) - *cane* (0.6%) - *verde* (0.6%) - *letto* (0.6%) - *divano* (0.6%) - *gatto* (0.6%) - *ragazza* (0.6%) - *rosso* (0.5%) - *cavallo* (0.5%) - *autobus* (0.4%) - *colore* (0.4%) - *piccolo* (0.4%) - *scrivania* (0.4%) - *grande* (0.4%) - *arancione* (0.4%) - *giallo* (0.4%) - *ripiano* (0.4%) - *elefante* (0.4%) - *persone* (0.4%) - *cappello* (0.4%) - *camicia* (0.3%) - *armadio* (0.3%) - *strada* (0.3%) - *bambino* (0.3%) - *treno* (0.3%) - *camion* (0.3%) - *legno* (0.3%) - *campo* (0.3%) - *metallo* (0.3%) - *laptop* (0.3%) - *giacca* (0.3%) - *giraffa* (0.3%) - *giocatore* (0.3%) - *signora* (0.3%)

Table 1: The 50 most frequent answers in the datasets. For each word the percentage of associated questions is reported.

Finally, to provide a valuable resource for real-scale evaluation of NLP systems, we manually validated a subset of the test material, by correcting $3,000$ question/answer pairs, randomly selected to preserve data balance. In particular, we also restored the gender inflection, lost during the previous normalization process.

The resulting dataset, namely **GQA-it**[7] is a large scale (possibly noisy) dataset made of more than 1.08 M of question/answers insisting on more than 80k images, with a test set partially validated. Specific statistics about GQA-it are reported in Table 2. Note that "silver" refers to non-validated material, while "gold" refers to manually validated ones. Each question/answer pair is connected to an image and the identifiers are aligned to the original GQA resource, thus enabling the reuse of further levels of valuable information, such as the knowledge graph associated with each image. Figure 1 shows both English and Italian Question Answer pairs for an example image taken from GQA-it.

Dataset	**#images**	**#quest./ans. pairs**
`train`	72,140	943,000
`valid`	10,234	132,062
`test-dev` (**silver**)	398	12,578
`test-dev` (**gold**)	398	3,000

Table 2: Statistics of The GQA-it dataset. The gold `test-dev` is a subset of the silver one.

Figure 1: Examples from the GQA-it dataset (image id n90294):
Q(A)$_{en}$: Is the remote to the right or to the left of the book? (right). **Q(A)**$_{it}$: *Il telecomando è a destra o a sinistra del libro? (destra)*
Q(A)$_{en}$: How thick is the book to the left of the remote? (thick). **Q(A)**$_{it}$: *Quanto è spesso il libro a sinistra del telecomando? (spesso)*
Q(A)$_{en}$: What device is to the left of the calculator made of plastic? (charger). **Q(A)**$_{it}$: *Quale dispositivo si trova a sinistra della calcolatrice di plastica? (caricabatterie)*
Q(A)$_{en}$: What's the charger made of? (platic). **Q(A)**$_{it}$: *Di cosa è fatto il caricabatterie? (plastica)*
Q(A)$_{en}$: Are there any phones? (no). **Q(A)**$_{it}$: *Ci sono dei telefoni? (no).*

4 Experimental Evaluation

To assess the quality of the produced GQA-it dataset, we trained and evaluated a state-of-the-

[7] The resource is publicly available at `https://github.com/crux82/gqa-it`.

art VQA system over the automatically generated material and evaluated over the 3,000 manually validated test set. In particular, we evaluated LXMERT (Learning Cross-Modality Encoder Representations from Transformers) presented by Tan and Bansal (2019).[8] This neural architecture models the VQA problem by stacking three neural encoders: an object/relationship encoder encoding (which encodes the input images), a language encoder (which encodes the input questions) and a cross-modality encoder (that combines the above multimodal embeddings). In a nutshell, LXMERT extracts visual and linguistic information, combines them in the cross-modal encoder and applies a (linear) classifier that associates each image/question pair to one of the n possible answers considered in the dataset.

The object detector uses a Faster R-CNN model (Ren et al., 2015) built over the ResNet-101 backbone (He et al., 2015) and pre-trained on the Visual Genome dataset (Krishna et al., 2017) to encode salient area of the input images. The language encoder is implemented as a BERT based model (Devlin et al., 2019). In Tan and Bansal (2019) best results are obtained without using existing pre-trained BERT models: the weights of this encoder are randomly initialized and pre-trained (together with the weights of cross-modality encoder) using a dedicated large scale dataset. This is composed of image captions and related questions of about 9 millions sentences. This pre-training stage is implemented by defining 5 auxiliary tasks, e.g., the cross-modal alignment task ("does the sentence describes the image?"). Nonetheless, experimental results showed that good performances can be also obtained by adopting a pre-trained BERT model. In order to effectively train LXMERT over GQA-it, we replaced the specialized English model with a standard pre-trained BERT model, in particular, multilingual BERT (Pires et al., 2019), which is also available for Italian. We preserved the original object/relationship encoder (which is language independent) and randomly initialized the cross-modality encoder.

Performances are measured in terms of Accuracy, i.e., the percentage of questions that exactly received the correct answer. All experiments were conducted using the same parameters used in Tan and Bansal (2019) but we investigated up to 15 epochs in the fine-tuning. Results are reported in Table 3. To compare the effectiveness of LXMERT on English and Italian data, we selected the common subset of $3,000$ question/answer pairs in both languages. The task is extremely challenging: A system assigning random answers would achieve an accuracy of 0.05%. Considering that the dataset is quite imbalanced, a baseline system assigning the most frequent answer (here, "*yes*"/ "*sì*") achieves 17.6%. First, we applied the best model from Tan and Bansal (2019) (namely `en-pretrain`) that is pre-trained over the dedicated corpus: while it achieves 60.0% (almost the state-of-the-art) on the entire English `test-dev` dataset, it achieves 59.0% on this subset. Tan and Bansal (2019) show that performances drop to 56.2% when using the original pre-trained BERT, and the English multilingual counterpart here achieves 55.3%. This drop in performances confirms the findings of Tan and Bansal (2019) and represents a sort of upper-bound for the experiments in Italian, as all the above setups are not affected by the noise introduced in the training material of GQA-it.

	Model	Accur.
-	**baseline (most freq. answer)**	17.6%
en	**`LXMERT en-pretrain`**	59.0%
	`LXMERT bert-multi.`	55.3%
it	**`LXMERT en-pretrain` + MT**	47.1%
	`LXMERT bert multi.` + MT	44.8%
	`LXMERT-it` (gold ans.)	51.0%
	`LXMERT-it` (silver ans.)	52.6%

Table 3: Results of LXMERT and LXMERT-it on 3,000 questions of GQA and GQA-it.

In order to assess the value of the new Italian resource, we first evaluated a trivial workflow that re-used the above English models in an Italian setting (first two rows in the Italian section of Table 3). First, we automatically translated the Italian questions using Opus-NMT in English ($\text{mt}_{it\rightarrow en}$). Second, we applied the English LXMERT models (`en-pretrain` and `bert-multilingual`) to derive the English answers. Finally, we applied Opus-NMT to translate back answers to Italian ($\text{mt}_{en\rightarrow it}$), after pairing them with the questions, as discussed in the previous section (cf. Table 3, rows `LXMERT en-pretrain` + MT and `LXMERT bert multi.` + MT). Indeed, this trivial workaround achieved significant results, i.e., 47.1% and 44.8%. This drop is par-

[8] https://github.com/airsplay/lxmert

tially due to the $it \rightarrow en$ translation, as the performances of the `en-pretrain` model drops from 59.0% to 54.5% when applied to English questions derived via machine translation, while the `bert-multilingual` from 55.3% to 51.3%. We suppose that the language model of LXMERT is not robust to the noise induced by the NMT. The remaining performance drop is clearly due to the translation $en \rightarrow it$, mainly due to polysemy and the other phenomena discussed in the previous section.

Conversely, the model trained over GQA-it, namely LXMERT-it, achieves 51.0% accuracy, which improves the previous results and it is more in line with the results obtained with `bert-multilingual` in English. Evaluating LXMERT-it w.r.t. the answers generated with the proposed methodology (namely *silver* answers) raises the accuracy to 52.6%. A manual analysis of the differences reveals that they are mainly due to gender inflections (e.g., "*alto*" vs "*alta*", in English "*tall*"). Unfortunately, these cases will inevitably be misclassified by LXMERT-it since it only observed masculine forms during training (which were introduced during the initial normalization phase.

We performed a qualitative error analysis on a random sample of the test set (10%). We identified 6 main error classes. Overall, 44% of the questions produced a wrong answer. First of all, we can make some considerations on these errors. On the one hand, specific errors are due to the wrong identification of objects in the images. In this paper, we did not modify the visual component of the architecture, and therefore the corresponding errors could not be avoided. Many other errors may be attributed to issues related to the machine translation, and in general with the creation of a noisy system for visual question answering. In particular, some errors are critical for the correct comprehension of questions and answers, and in general for using the Italian VQA model. In fact, some errors compromise the correct understanding of the answers (e.g., "*right*" translated in Italian as "*corretto*" instead of "*destra*"), while others allow the correct (albeit noisy) use of the system, such as the use of synonyms and hypernyms of the gold class.

Error Type	**Example(s)**	**Perc.**
Object	*tavola* ('table') vs *sedia* ('chair')	31%
Synonyms or hypernyms	*persona* ('person') vs *donna* ('woman')	17%
Attributes	*blu* ('blue') vs *nero* ('black'); *chiuso* ('closed') vs *aperto* ('open')	14%
Morph. feat.	*bella* ('beautiful') vs *bello* ('beautiful'); *persona* ('person') vs *persone* ('people')	3%
Actions	*sta dormendo* ('sleeping') vs *sta sdraiato* ('is lying down)'	3%
Spatial feat.	*destra* ('right') vs *sinistra* ('left')	2%
Residual	*si* ('yes') vs *no* ('no')	31%

Table 4: Classification of errors in `LXMERT-it`.

5 Conclusions

This paper presents GQA-it, a collection of more than 1 M question/answer pairs in Italian associated to 80k images in support of research in VQA in Italian. GQA-it has been obtained with machine translation, and the quality of the resulting resource is demonstrated through both direct evaluation of the translation and indirect evaluation of a state-of-the-art model trained on this material.

This work represents a first step to leverage a large-scale VQA resource like GQA for Italian, a resource whose quality can still largely been improved. In particular, the knowledge graphs behind each image will be extremely valuable to improve the final resource (e.g., using a generation process as in (Hudson and Manning, 2019)) or the VQA process. Finally, the available alignment between GQA and GQA-it will foster research in cross-lingual VQA.

The aim of this paper was to explore the possibility of semi-automatically inducing large-scale Italian dataset for VQA. Obviously, we are aware that there is plenty of room for improvement in many respects. First, a wide range of approaches could be tested, aimed at reducing the noise due to the adaptation of English resources to Italian ones. Specifically, a viable option could be to leverage the question and the image together with each other in order to provide a more consistent translation. Finally, a multimodal masked language modeling step on text-image pairs could enrich the Italian BERT model and make it comparable with the English counterpart. We plan to probe these research avenues in the near future.

Acknowledgments

We would like to thank the "Istituto di Analisi dei Sistemi ed Informatica - Antonio Ruberti" (IASI) for supporting the experimentations through access to dedicated computing resources.

References

Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. 2018. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4971–4980.

Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. 2016. Neural module networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 39–48.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Kan Chen, Jiang Wang, Liang-Chieh Chen, Haoyuan Gao, Wei Xu, and Ram Nevatia. 2015. Abc-cnn: An attention based convolutional neural network for visual question answering. *arXiv preprint arXiv:1511.05960*.

Danilo Croce, Alexandra Zelenanska, and Roberto Basili. 2019. Enabling deep learning for large scale question answering in italian. *Intelligenza Artificiale*, 13(1):49–61.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913.

Deepak Gupta, Pabitra Lenka, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A unified framework for multilingual and code-mixed visual question answering. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 900–913, Suzhou, China, December. Association for Computational Linguistics.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition. *CoRR*, abs/1512.03385.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Mateusz Malinowski and Mario Fritz. 2014. A multi-world approach to question answering about real-world scenes based on uncertain input. *Advances in neural information processing systems*, 27:1682–1690.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy, July. Association for Computational Linguistics.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Yash Srivastava, Vaishnav Murali, Shiv Ram Dubey, and Snehasis Mukherjee. 2020. Visual question answering using deep learning: A survey and performance analysis.

Hao Tan and Mohit Bansal. 2019. LXMERT: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, Hong Kong, China, November. Association for Computational Linguistics.

Damien Teney, Peter Anderson, Xiaodong He, and Anton Van Den Hengel. 2018. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4223–4232.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. 2016. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 21–29.

Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. 2016. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4995–5004.

Implementing a Pragmatically Adequate Chatbot in DialogFlow CX

Anna Dall'Acqua[1,2], **Fabio Tamburini**[1]
1. FICLIT, University of Bologna, Italy
2. Injenia S.r.l., Bologna, Italy
anna.dallacqua2@unibo.it, fabio.tamburini@unibo.it

Abstract

This paper presents work in progress concerning the implementation of a list of linguistic patterns developed in an original way to be pragmatically adequate. These patterns for Italian are strongly rooted in Conversation Analysis and are adaptable and portable into different domains. The platform used for the implementation is Dialogflow CX.

1 Introduction

Although the first dialogue systems began to appear around the second half of the last century (Weizenbaum, 1966; Colby et al., 1971) is it especially in recent years that we have witnessed a proliferation of these technologies in a wide variety of fields (Tsvetkova et al., 2017; Chaves et al., 2019; Dale, 2016). The numerous attempts that have been made to classify them (Radziwill and Benton, 2017; Følstad et al., 2019; Hussain et al., 2019; Mathur and Sing, 2018) and the absence of an unequivocal taxonomy (Braun and Matthes, 2019) contribute to the lack of a methodological approach for designing conversational agents.

The recent technological developments have led to the standardisation of the technical frameworks: the main Natural Language Understanding (NLU) platforms, both developed by technology giants such as Google Dialogflow, IBM Watson, and Microsoft Luis and those from the open source community such as RASA, contributed to the affirmation of the dominant paradigm based on *intents*, *entities* and *responses* for building conversational agents (Adamopolou and Moussiades, 2020; Moore and Arar, 2019). The existing flourishing literature about this aspect (Ahmad et al., 2018; Adamopolou and Moussiades, 2020) has not been associated with equivalent research on methods and linguistic theories that can be pursued for the design phase of conversational projects. During the survey of methodological studies on conversation design, it became clear that there is no shared standard and that various methodological contributions of a practical nature do not refer to a specific theoretical linguistic perspective (Dasgupta, 2018; Pearl, 2016; Cohen et al., 2004; Hall, 2018).

In this work we embrace the Natural Conversation Framework (NCF) whose validity has been already demonstrated in Dall'Acqua and Tamburini (in press); we select some of its most representative patterns and we implemented them on the newly released version of Google Dialogflow CX. This paper is intended as a continuation of the work presented in Dall'Acqua, Tamburini (in press), which sets out the theoretical and methodological assumptions on which this work is based.

2 The *Natural Conversation Framework* as a Theoretically Funded Approach

Among the linguistic approaches available to analyse interactional exchanges, a *pragmatic perspective* appears to be the most appropriate (Bianchini et al., 2017), especially in its declination of *Conversation Analysis* (Schegloff et al., 1977; Sacks et al., 1974). For this reason, we claim that the *Natural Conversation Framework (NCF)* identified by Moore and Arar (2019), consisting of language patterns structured into sequences in the theoretical groove of Conversation Analysis, could be a promising starting point for the definition of a potentially generalisable and adaptable linguistic methodology.

Since we have already demonstrated the theoretical validity of this approach and we have included it in a practical and applicative procedural workflow on Dialogflow ES (Dall'Acqua and Tamburini, in press) this work aims to continue the research by transposing some of the most significant patterns on the new and very recent version (Nov. 2020) of the platform.

3 Dialogflow CX

The renewed version of Dialogflow is linked to the *information-based* approach (Larsson and Traum, 2000; Traum and Larsson, 2003) and opens to more dynamic scenarios: since it is structured as a finite-state machine, it allows the users to build more flexible, reusable and adaptable patterns. The level of dialogues complexity that can potentially be created is enhanced by the wider range of features that the new tool has to offer: it allows the transition from one state of the conversation to another to be visualized through the creation of *pages*, which are the states of the underlying state machine, configured to collect end-user information relevant to that state of the conversation[1]. The conversational flow itself is therefore made of pages, connectors between the pages (known as *state handlers*[2]) and *flows*, reciprocally independent units of dialogues used to manage more complex conversational agents.

4 Conversational Architecture and Pattern Selection

We enlarged the implementation started in our previous work combining together in an original way a selection of patterns identified by Moore and Arar (2019) and trying to reproduce the most representative, widespread and generalisable use-cases of an high-level conversational agent with practical purposes roughly oriented to customer care. Here, it is not relevant the precise use of the demonstrated chatbot, as the main point is to show and describe the potentialities of the proposed approach. We have adapted patterns taken to all the categories of the classification proposed by Moore and Arar (2019) and suggest that they may also be considered as a best-practice to be taken into account in the summary roadmap towards the implementation previously presented.

In our work we have created three flows. As Fig. 1 shows, they are all connected to the main flow *(Default Start Flow)* both in the initial and in the final state of each flow. The three flows are:

[1]https://cloud.google.com/dialogflow/cx/docs/concept/page

[2]https://cloud.google.com/dialogflow/cx/docs/concept/handler

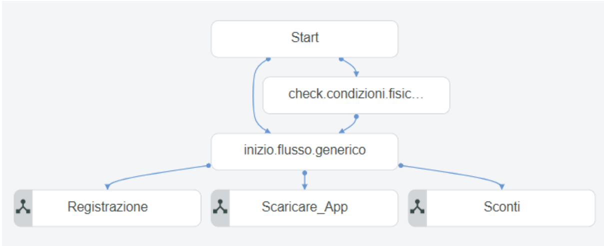


Figure 1: Overall flows architecture.

- **Registration** (*Registrazione*): it reproduces an online registration procedure. It aims to generalise the use-case in which the user has to provide some data (*entities*), divided into mandatory data (without which the procedure cannot succeed) and optional data (the registration can take place correctly even without these data). The procedure of extracting data from the user (*slot filling)* is portable to multiple domains (Mohamad Suhaili et al., 2021) (Fig. 2).

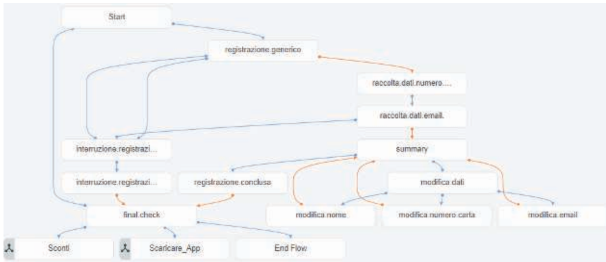


Figure 2: Registration flow diagram.

- **App Download** (*Scaricare App)*: it supports the user during a download procedure in multiple steps. It aims to show the application of the *story-telling sequences* (Jefferson, 1978) used to express a content that needs to be parcelled out into smaller pieces of speech. Furthermore, it offers a rudimentary troubleshooting procedure in case of error during the download, that can be actually used to diagnose and manage also other typologies of errors. The widespread of troubleshooting procedures in chatbots is demonstrated in (Thorne, 2017), which also endorses the portability of this type of conversational interactions into multiple domains (Fig. 3).

- **Discounts** (*Sconti)*: this flow is dedicated to typologies of discounts and promotions available and it is used to show a combination of patterns that allows to manage series of contextual questions related to the same subject (Fig. 4).

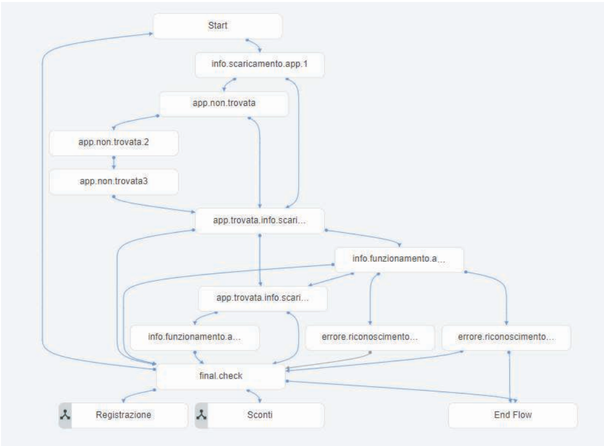


Figure 3: App Download flow diagram.

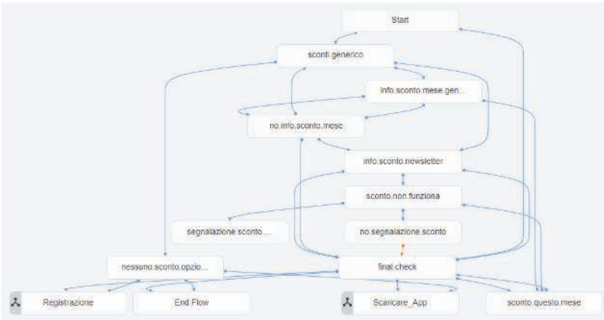


Figure 4: Discounts flow diagram.

Now we are going to review the most pragmatically relevant points of each created flow, aiming to show the adaptability of the patterns also to the Italian language and its suitability to multiple context of use.

4.1 Default Start Flow

The Default Start Flow opens with a slightly revisited version of the pattern **C1.4 Opening Welfare Check (Agent)**[3]: if the `NAME REQUEST` in the first turn is for some reasons not satisfied, the agent goes straight to the practical purposes of the conversation, skipping the `WELFARE CHECK`, as the following dialogues show:

1 A: Ciao, sono Teo, il tuo assistente virtuale. Come ti chiami?
2 U: Anna, piacere.
3 A: Piacere di conoscerti, Anna. Come stai?
4 U: al top
5 A: Ottimo, sono felice per te! Sono qui per darti qualche informazione su alcuni semplici argomenti: gli sconti e la procedura di scaricamento della nostra app. Inoltre, posso aiutarti a registrarti sul nostro sito, direttamente in chat. Di che cosa hai bisogno?

1 A: Ciao, sono Teo, il tuo assistente virtuale. Come ti chiami?
2 U: cavoli miei
3 A: Ok, non importa, intanto sai già il mio :) Sono qui per darti qualche informazione su alcuni semplici argomenti: gli sconti e la procedura di scaricamento della nostra app. Inoltre, posso aiutarti a registrarti sul nostro sito, direttamente in chat. Di che cosa hai bisogno?

The agent's reaction in the turn `WELFARE RECEIPT` (`NEUTRAL`/`POSITIVE`/`NEGATIVE`) is also conditioned by the user's response. From the page **inizio.flusso.generico** *(generic starting flow)* the intents that introduce the flows can be activated, or the user can freely leave the conversation if he/she is not interested in the proposed subjects. In addition, we have also implemented a new version of the pattern **B4.2 Sequence closer Appreciation (helped)**: instead of an admission of not-knowing from the chatbot (`NO ANSWER ACCOUNT`), we have collocated at one of the initial pages, before that one flow has ever being activated: this condition motivates the answer provided in the following dialog after the thanks from the user:

1 A: Ciao, sono Teo, il tuo assistente virtuale. Come ti chiami?
2 U: Ciao sono Francesco
3 A: Piacere di conoscerti, Francesco. Come stai?
4 U: bene
5 A: Bene. Sono qui per darti qualche informazione su alcuni semplici argomenti: gli sconti e la procedura di scaricamento della nostra app. Inoltre, posso aiutarti a registrarti sul nostro sito, direttamente in chat. Di che cosa hai bisogno?
6 U: Grazie mille
7 A: Non mi pare di aver fatto molto... Tuttavia, prego!

The same pattern is handled differently in the other flows, if the agent actually provided help to the user.

4.2 Registration *(Registrazione)*

When the user directly types something like "I'd like to register to the site" (*Vorrei registrarmi sul sito)*, the dedicated flow will be activated and the data extraction procedure will be started. The most appropriate patterns to extract data are: **A2.6 Open Request Summary**, **A2.7 Warrant Request & Refusal**, **A2.11 Open Request Repairs** and **B2.4. Paraphrase Request (elaboration)** combined in original way in order to manage both

[3]The complete list of all patterns is available in Moore and Arar (2019) and on the website `https://researcher.watson.ibm.com/researcher/view_group_subpage.php?id=10182`. The Appendix lists the implemented patterns.

mandatory and optional data to complete the online registration. Similar patterns have been already implemented in Dialogflow ES (Dall'Acqua and Tamburini, in press), but the output on Dialogflow CX is even more effective also in terms of feasibility of the implementation. As the following dialogue shows

1	U:	Volevo registrarmi al sito
2	A:	Per poterti registrare al sito e creare un tuo profilo devo raccogliere qualche tuo dato. Avrei bisogno di nome e cognome, come ad esempio: "Marco Rossi".
3	U:	Lucia Mondella
4	A:	Qual è il tuo numero di carta fedeltà?
5	U:	Quale carta?
6	A:	La carta fedeltà è una carta che ti permette di raccogliere punti ad ogni tuo acquisto, in negozio e online. Non è obbligatoria, ma attivandola potrai godere di sconti speciali su diversi nostri prodotti. Qual è il tuo numero di carta fedeltà?
7	U:	Quale numero ti devo dire?
8	A:	Il codice è a 5 cifre e si trova sul retro della tua carta fedeltà.
9	U:	non lo vedo
10	A:	Il codice di carta fedeltà è a cinque cifre (es: "12345") e si trova sul retro della carta, in basso a destra. Qual è il tuo numero di carta fedeltà?
11	U:	non ce l'ho
12	A:	Ok, non c'è problema. In ogni caso, potrai aggiornare questa sezione in ogni momento della pagina personale del tuo profilo. Per finire, ho bisogno di un'ultima informazione. Qual è la tua email?
13	U:	luciamondella@libero.it
14	A:	Perfetto. Ricapitolando: avvio la procedura di registrazione a nome di Lucia Mondella, numero di carta non fornito, e invio un'email di conferma all'indirizzo luciamondella@libero.it. È tutto corretto?
15	U:	no, volevo inserire un dato
16	A:	quale dato volevi inserire?
17	U:	ho recuperato la carta.
18	A:	Certamente. Ridigita pure il numero della carta fedeltà. Ti ricordo che è un numero a cinque cifre e si trova sul retro della carta.
19	U:	12345
20	A:	Perfetto. Ricapitolando: avvio la procedura di registrazione a nome di Lucia Mondella, numero di carta 12345, e invio un'email di conferma all'indirizzo luciamondella@libero.it. È tutto corretto?
21	U:	Ottimo, Lucia Mondella. Abbiamo finito. A breve, riceverai una mail all'indirizzo che mi hai indicato per il completamento della registrazione. Hai bisogno di altre informazioni su sconti, app o registrazione?
22	A:	no, son a posto
23	U:	Va bene. Alla prossima!

we have adapted the selected patterns to the situation, transposing them into questions related to the topic "fidelity card" like "which card?" (*quale carta?*) and "which number should I say?" (*quale numero devo dire?*) and a repair procedure with "I can't see it" (*non lo vedo*). It is noticeable that in case of questions or repairs related to it that are predictable, the agent reacts accordingly paraphrasing what it has previously said or providing additional information that may help the user to supply the requested data, without stopping to prompt him/her to move the conversation forward ("Which is your fidelity's card number?" *Qual è il numero della tua carta fedeltà?*). The main difference between the same conversational flow on the two versions of Dialogflow is the effectiveness of the final summary, which is a requirement for a correct data extraction procedure (Moore and Arar, 2019). Here the user can choose to not provide an optional data such as the fidelity card number, but if after the final summary he/she chooses instead to insert it, the agent allows the modification and shows the restored value in the next `REQUEST SUMMARY`, asking again if the summary is correct or not. If is it correct, this flow is going to an end and it proposes one of the last pages of it, namely the **final.check.** realised with the pattern **C4.0 Last Topic Check (Agent)** in which the agent invites the user or to transit from another flow, or to freely interrupt the conversation. Another relevant ramification of the flow "Registrazione" shows the management of an obligatory data, such as the name:

1	A:	Mi devo registrare al sito
2	U:	Per poterti registrare al sito e creare un tuo profilo devo raccogliere qualche tuo dato. Avrei bisogno di nome e cognome, come ad esempio: "Marco Rossi".
3	U:	perché?
4	A:	Il nome è un requisito essenziale per la registrazione. Non preoccuparti, sarà gestito come dato personale e sarà visibile solo a te. Come ti chiami?
5	U:	non voglio dire il mio nome ad un robot
6	A:	Capisco. Senza questo dato, però, non posso aiutarti con la registrazione. Preferisci interrompere qui?
7	U:	no va bene
8	A:	Va bene. Ripren diamo allora da dove eravamo rimasti. Per poterti registrare al sito e creare un tuo profilo devo raccogliere qualche tuo dato. Avrei bisogno di nome e cognome, come ad esempio: "Marco Rossi".
9	U:	Stupido robot
10	A:	Non serve a nulla reagire in modo aggressivo. Come ti chiami?
11	U:	cretino!
12	A:	In questo modo stai ferendo il mio povero cuore metallico. Come ti chiami?
13	U:	non te lo dico
14	A:	Capisco. Senza questo dato, però, non posso aiutarti con la registrazione. Preferisci interrompere qui?
15	U:	sì ciao

Also in this case we have provided a series of specific question related to the topic such as "why?" (*perché?*), but if the user refuses to provide this data, the chatbot suggests to interrupt the procedure. Since the goal of this demo is to show the potentialities of this methodology, we have stressed the possibilities of the conversation making the user going back to the previous procedure, apparently willing to provide the name. But once the data extraction procedure starts again, the user starts to insult the robot, that reacts accordingly to every insult but without stopping to ask for the name, until the user communicates his/her will of interrupting the procedure and the conversation.

4.3 App Download *(Scaricare App)*

We have figured this flow out to show the transponibility of patterns **A3.0 Extended Telling with Repair** and **A3.1 Extended Telling Abort**, that are normally used to expose a story, an anecdote or a list of instructions that cannot be achieved in one single conversational turn. This flow can be activated taping an expression associated to the corresponding intent, such as “how do I download the app?” (*come scarico l’app?*), that leads to the first step of the explanation, with an extra-prompt to the user (“Did you find it?” *L’hai trovata?*) to move the conversation forward. If the user claims he/she did not find it, the agent provides additional information to make the recognition easier, which may eventually end up in troubleshooting procedure carried out by the pages **“app.non.trovata”, “app.non.trovata.2”, “app.non.trovata.3”**.

1	U:	come si scarica l’app?
2	A:	Prima di tutto, vai su App Store o su Playstore e digita il nome della nostra catena nella barra di ricerca. Così facendo dovresti vedere la nostra app. L’hai trovata?
3	U:	Non la vedo
4	A:	L’app si chiama esattamente come la nostra catena di negozi ed è rappresentata da un triangolo di colore verde. Prova a uscire rientrare dall’App Store che stai utilizzando e a digitare nuovamente il nome nella barra di ricerca. Adesso l’hai trovata?
5	U:	eccola
6	A:	Perfetto, adesso puoi scaricarla. Ci vorrà solo qualche secondo.
7	U:	come si fa?
8	A:	Facendo tap su “installa” dovresti riuscire a scaricare l’app sul tuo smartphone.
9	U:	ok
10	A:	A questo punto, una volta installata, ti basta inserire la tua mail e il tuo numero della carta fedeltà per associare la tua app al tuo profilo online, aggiornare tutte le tue informazioni e poter fare acquisti.
11	U:	grazie mille
12	A:	Prego! Hai bisogno di altre informazioni su sconti, app o registrazione?

The last two pages represent the final states of troubleshooting, which therefore may not be invoked if the user has solved the problem in the previous states.

The conversation is articulated in a sequence of instructions: each following step in the explanation is motivated by the `CONTINUER` tipped by the user, that allows the agent to provide another piece of information until the procedure is finished (Schegloff, 1982). The conversational turns in which the user simply reports that the previous step has been received can be alternated with repair procedures, in which the user asks for explanations regarding a specific step (Sacks et al., 1992), as shown in the dialogue above. In every moment of the procedure, the user may also notify that he/she has actually completed the procedure through an intent categorised as `COMPLAINT/ABORT REQUEST` and the agent should therefore stop providing instructions. In our flow the agent acknowledges the user’s request (”Great!” *Ottimo!*) and steps forward to the page **final.check.**:

1	U:	come faccio a scaricare l’app?
2	A:	Prima di tutto, vai su App Store o su Playstore e digita il nome della nostra catena nella barra di ricerca. Così facendo dovresti vedere la nostra app. L’hai trovata?
3	U:	sì
4	A:	Perfetto, adesso puoi scaricarla. Ci vorrà solo qualche secondo.
5	U:	Ok basta ho risolto grazie
6	A:	Ottimo! Hai bisogno di altro su sconti, app oppure registrazione?

4.4 Discounts *(Sconti)*

With this last flow we would like to demonstrate the applicability of two more patterns, namely the pattern **A2.9 Open Request Incremental A2.10 Open Request Series**, **A2.4 Open Request Agent Detail Request**. They are used to manage situations in which the users asks a series of questions related to a single topic or switches topic but asks similar questions related to it. As the following dialogue shows

1	U:	Che sconti avete?
2	A:	Esistono due principali tipologie di sconto: lo sconto newsletter e lo sconto del mese. Su quale di queste vorresti ricevere maggiori informazioni?
3	U:	il primo
4	A:	Iscrivendoti alla nostra newsletter riceverai uno sconto di del 10% utilizzabile una sola volta su una spesa di almeno 30 euro, online o in negozio.
5	U:	Posso usarlo anche per un acquisto online?
6	A:	Non ho capito, puoi ripetere la tua domanda sullo sconto newsletter?
7	U:	posso usare lo sconto newsletter sia online sia in negozio?

8 A: Puoi utilizzare lo sconto newsletter su tutti i tuoi acquisti, sia online, sia in negozio.
9 U: Come funziona?
10 A: Basta cliccare su “Iscrizione newsletter” in alto a destra in ogni pagina del sito e inserire la tua email nella tab che si apre. Presto arriverà in posta un codice sconto a sei cifre che potrai applicare ad un acquisto online o in negozio a partire da 30 euro di spesa.
11 U: ok
12 A: Hai bisogno di altro sugli sconti, sulla procedura di registrazione, o su come scaricare l’app?
13 U: volevo delle informazioni sullo sconto del mese
14 A: Per il mese di gennaio, prevediamo una promozione sugli articoli per bambini, con uno sconto del 50% su ogni prodotto indicato con il simbolo “Child” sia online sia in negozio.
15 U: Quando scade?
16 A: Questo tipo di sconto si rinnova di mese in mese: questo, ad esempio, sarà valido fino al 31 di gennaio.
17 U: come si usa?
18 A: Lo sconto verrà automaticamente applicato se il prodotto scelto per l’acquisto è oggetto della promozione: se stai acquistando online, verrà applicato il prezzo scontato indicato accanto al protocollo, altrimenti ti verrà applicato lo sconto direttamente in cassa.
19 U: ok grazie mille
20 A: Prego! Hai bisogno di altro sugli sconti, sulla procedura di registrazione, o su come scaricare l’app?
21 U: no grazie
22 A: Va bene. Alla prossima!

the conversation opens with a generic open request from the user (“what kinds of discounts do you have?” *che sconti avete?*) that is satisfied by the agent with a request of more details: the agent needs to know the specific type of discount as additional detail to provide specific information about it. Once the type of discount is defined, the user can start asking specific question related to it without always specifying the subject. Once the user has satisfied his/her needs in relation to newsletter discount, at the **final.check** page he/she can switch the topic and start asking a series of question related to the other one, until the user has achieved all the needed information. Once the user thanks the agent, this is perceived as an acknowledgement of a successful conversation, so the conversational flow can go away.

5 Conclusions and Future Directions of the Research

We have demonstrated the applicability of this method also on the new released version of one of the most important Natural Language Understanding platform, namely Dialogflow CX. Since this version of Dialogflow has been released for the Italian language only in November 2020, to our knowledge this is the only study in which this platform is used for Italian and for the realisation of a conversational project with practical purposes. This is therefore a further and more complete implementation of the pragmatic adequateness of this approach.

In the future, we would like to evaluate the effectiveness of this approach on a fully functional prototype that can be adapted not only for commercial purposes, but also for other important application contexts, such as education. Due to the variety of frameworks available for the evaluation process (Casas et al., 2020), the question of the most suitable evaluation method is still open for further discussion.

Acknowledgements

We would like to thank Injenia S.r.l. for supporting this research. CRediT author statement; ADA: Conceptualization, Methodology, Validation, Formal Analysis, Investigation, Writing (Original Draft), Writing (Review & Editing); FT: Conceptualization, Supervision, Project Administration, Writing (Review & Editing), Funding Acquisition.

References

E. Adamopolou and L. Moussiades. 2020. An overview of chatbot technologies. *Artificial Intelligence Applications and Innovations*, 584:373–383.

N. A. Ahmad, M. H. Che, A. Zainal, M. A. R. Fairuz, and Z. Adnan. 2018. Review of chatbots design techniques. *International Journal of Computer Applications*, 181(8):7–10.

A. Bianchini, F. Tarasconi, R. Ventaglio, and M. Guadalupi. 2017. “gimme the usual” - how handling of pragmatics improves chatbots. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017)*, pages 30–35.

D. Braun and F. Matthes. 2019. Towards a framework for classifying chatbots. In *Proceedings of the 21st International Conference on Enterprise Information Systems (ICEIS 2019)*, pages 484–489. volume 1.

J. Casas, M.-O. Tricot, O. Abou Khaled, E. Mugellini, and P. Cudré-Mauroux. 2020. Trends & methods in chatbot evaluation. In *Companion Publication of the 2020 International Conference on Multimodal Interaction*, page 280–286, New York, NY, USA. Association for Computing Machinery.

A. P. Chaves, E. Doerry, J. Egbert, and M. Gerosa. 2019. It’s how you say it: Identifying appropriate register for chatbot language design. In *Proceedings of the 7th International Conference on Human-Agent Interaction*, pages 102–109.

M. H. Cohen, J. P. Giangola, and J. Balogh. 2004. *Voice User Interface Design*. Addison Wesley.

K. M. Colby, S. Weber, and F. D. Hilf. 1971. Artificial paranoia. *Artificial Intelligence*, 2(1):1–25.

R. Dale. 2016. Industry watch the return of chatbots. *Natural Language Engineering*, 22(5):811–817.

A. Dall'Acqua and F. Tamburini. in press. Toward a linguistically grounded dialog model for chatbot design. *Italian Journal of Computational Linguistics*, 7(1).

R. Dasgupta. 2018. *Voice User Interface Design. Moving from GUI to Mixed Modal Interaction*. Apress.

A. Følstad, M. Skjuve, and P. B. Brandtzaeg. 2019. Different chatbots for different purposes: Towards a typology of chatbots to understand interaction design. In S. S. Bodrunova, O. Koltsova, A. Følstad, H. Halpin, P. Kolozaridi, L. Yuldashev, A. Smoliarova, and H. Niedermayer, editors, *Internet Science. INSCI 2018. Lecture Notes in Computer Science*, pages 145–156. vol 11551, Springer, Cham.

E. Hall. 2018. *Conversational design*. A Book Apart.

S. Hussain, O. Ameri Sianaki, and N. Ababneh. 2019. A survey on conversational agents/chatbots classification and design techniques. In L. Barolli, M. Takizawa, F. Xhafa, and T. Enokido, editors, *Web, Artificial Intelligence and Network Applications. WAINA 2019. Advances in Intelligent Systems and Computing*. vol 927. Springer, Cham.

G. Jefferson. 1978. Sequential aspects of storytelling in conversation. In J. Schenkein, editor, *Studies in the Organization of Conversational Interaction*, pages 219–48. Academic Press.

S. Larsson and D.R. Traum. 2000. Information state and dialogue management in the trindi dialogue move engine toolkit. *Natural Language Engineering*, 6(3-4):323–340.

V. Mathur and A. Sing. 2018. The rapidly changing landscape of conversational agents. *arXiv 1803.08419v2*, pages 1–14.

S. Mohamad Suhaili, N. Salim, and M. Nazim Jambli. 2021. Service chatbots: A systematic review. *Expert Systems with Applications*, 184:115461.

R. J. Moore and R. Arar. 2019. *Conversational UX Design. A Practitioner's Guide to the Natural Conversation Framework*. Association for Computing Machinery.

C. Pearl. 2016. *Designing Voice User Interfaces. Principles of Conversational Experiences*. O'Reilly.

N. Radziwill and M. Benton. 2017. Evaluating quality of chatbots and intelligent conversational agents. *arXiv 1704.04579*.

H. Sacks, E. A. Schegloff, and G. Jefferson. 1974. A simplest systematics for organization of turn-taking for conversation. *Language*, 50:696–735.

H. Sacks, E. Schegloff, and G. Jefferson. 1992. *Lectures on Conversation*, volume I, II. Blackwell.

E. A. Schegloff, G. Jefferson, and H. Sacks. 1977. The preference for self-correction in the organization of repair in conversation. *Language*, 53(2):361–382.

E.A. Schegloff. 1982. Discourse as an interactional achievement: Some uses of 'uh huh' and other things that come between sentences. In D. Tannen, editor, *Analysing Discourse: Text and Talk*, page 71–93. Washington DC, Georgetown University Press.

C. Thorne. 2017. Chatbots for troubleshooting: A survey. *Language and Linguistics Compass*, 11(10):e12253. e12253 LNCO-0688.R1.

D.R. Traum and S. Larsson. 2003. The information state approach to dialogue management. In J. van Kuppevelt and R.W. Smith, editors, *Current and New Directions in Discourse and Dialogue*, pages 325–353. Springer Netherlands, Dordrecht.

M. Tsvetkova, R. García-Gavilanes, L. Floridi, and T. Yasseri. 2017. Even good bots fight: The case of wikipedia. *PLOS ONE*, 12(2):1–27.

J. Weizenbaum. 1966. Eliza – a computer program for the study of natural language communication between man and machine. *CACM*, 9(1):36–45.

Appendix

List of the implemented patterns:

C1.4 Opening Welfare Check (Agent)

1	A:	GREETING. SELF-IDENTIFICATION. NAME REQUEST.
2	U:	NAME.
3	A:	GREETING, DIRECT ADDRESS. WELFARE CHECK.
4	U:	WELFARE REPORT (NEUTRAL / POSITIVE/ NEGATIVE)
5	A:	WELFARE RECEIPT (NEUTRAL / POSITIVE / NEGATIVE)

B4.3 Seq. Closer Appreciation (not helped)

1	A:	INQUIRY/REQUEST
2	U:	NO ANSWER ACCOUNT
3	A:	APPRECIATION
4	U:	REFUTATION

A2.6 Open Request Summary

1	U:	PARTIAL REQUEST
2	A:	DETAIL REQUEST
3	U:	DETAIL
4	A:	DETAIL REQUEST
5	U:	DETAIL
6	A:	DETAIL REQUEST
7	U:	DETAIL
8	A:	REQUEST SUMMARY
9	U:	SUMMARY CONFIRM
10	A:	GRANT
11	U:	SEQUENCE CLOSER
12	A:	RECEIPT

A2.7 Warrant Request & Refusal

1 A: DETAIL REQUEST
2 U: WARRANT REQUEST
3 A: WARRANT
4 U: REFUSAL
5 A: ACKNOWLEDGEMENT. <NEXT SECTION>

A2.11 Open Request Repairs

1 U: FULL REQUEST
2 A: GRANT
3 U: REPAIR INITIATOR
4 A: REPAIR
5 U: SEQUENCE CLOSER
6 A: RECEIPT

B2.4 Paraphrase Request (elaboration)

1 U: <ANY UTTERANCE>
2 A: PARAPHRASE REQUEST
3 U: PARAPHRASE DEFAULT

A3.0 Extended Telling with Repair

1 A: STORY/INSTRUCTION INVITATION
2 U: PART/STEP 1
3 A: CONTINUER/PAUSE
4 U: PART/STEP 2
5 A: REPAIR INITIATOR
6 U: REPAIR
7 A: CONTINUER/PAUSE
8 U: PART/STEP 3
9 A: SEQUENCE CLOSER
10 U: RECEIPT

A3.1 Extended Telling Abort

1 A: STORY/INSTRUCTION INVITATION
2 U: PART/STEP 1
3 A: CONTINUER/PAUSE
4 U: PART/STEP 2
5 A: REPAIR INITIATOR
6 U: REPAIR
7 A: PART/STEP 3
8 U: COMPLAINT/ABORT REQUEST
9 A: ABORT OFFER
10 U: ABORT CONFIRM
11 A: ACKNOWLEDGEMENT

A2.9 Open Request Incremental

1 U: FULL REQUEST
2 A: GRANT
3 U: INCREMENTAL REQUEST
4 A: GRANT
5 U: SEQUENCE CLOSER
6 A: RECEIPT

A2.10 Open Request Series

1 U: FULL REQUEST
2 A: GRANT
3 U: RELATED REQUEST
4 A: GRANT
5 U: SEQUENCE CLOSER
6 A: RECEIPT

A2.4 Open Request Agent Detail Request

1 U: PARTIAL REQUEST
2 A: DETAIL REQUEST
3 U: DETAIL
4 A: GRANT
5 U: SEQUENCE CLOSER
6 A: RECEIPT

La produzione di testi semplificati di notiziari televisivi italiani destinati a persone con disturbi cognitivi acquisiti: un'integrazione tra metodi psicolinguistici e analisi automatiche

Maria De Martino[1], **Annalisa Colella** [1,2]
1. University of Salerno, Salerno, Italy
2. RAI, Rome, Italy
mdemartino@unisa.it, annalisa.colella@rai.it

Abstract

English. The goal of the study was to implement a linguistically and cognitively-oriented procedure aimed at improving the quality of use and comprehension of TV contents for an audience of people suffering from cognitive disorders after a brain damage.
Tools for the automatic text analysis and psycholinguistic and neuropsychological methods have been exploited in order to obtain simplified versions of original written texts from Italian TV news bulletins. An empirical pilot study on healthy people has been conducted where reading latencies for original vs. simplified texts have been compared.

Italiano. *Il lavoro ha lo scopo di mettere a punto una procedura cognitivamente e linguisticamente orientata per migliorare la fruizione di contenuti televisivi da parte di persone con disturbi cognitivi conseguenti a lesioni cerebrali. Attraverso l'integrazione di tecniche di analisi linguistica automatica e di metodi provenienti dalla psicolinguistica e dalla neuropsicologia sono state ottenute versioni semplificate dei testi usati nei notiziari televisivi italiani. È stato condotto uno studio empirico pilota con parlanti sani in cui è stata valutata la velocità di lettura dei testi originali e semplificati.*

1 Introduzione

In Italia e nel mondo una parte importante della popolazione sopravvive a lungo ad eventi traumatici e patologici a carico del cervello che compromettono in maniera più o meno grave le capacità cognitive (Di Luca et al., 2011).
Il progresso nelle cure e nelle politiche sociali garantisce alle persone con lesioni cerebrali una buona aspettativa di vita e delle adeguate condizioni di salute generale.
Tuttavia, la qualità della vita non è paragonabile al periodo precedente all'insorgere del disturbo. Una delle ragioni che incide sulle ricadute negative a lungo termine di una lesione cerebrale ha a che fare con la riduzione della capacità delle persone di mantenersi orientate rispetto alla realtà socio-culturale in cui vivono, ad esempio, aggiornandosi attraverso i mezzi di informazione (Cartwright & Elliott, 2009; Denicolai, 2016).
Attualmente, le reti nazionali RAI, nell'ambito dei servizi di Pubblica Utilità, forniscono supporti per il pubblico ipovedente e non vedente (audio-descrizione), per il pubblico non udente (traduzioni in lingua italiana dei segni (LIS) e sottotitolazione) e altri strumenti generici di *Stretch TV* per adattare la velocità audio-video alle esigenze dell'utente. Tuttavia, non sono presenti servizi di supporto alle edizioni dei telegiornali (TG) specificamente destinati a persone con danni cognitivi. Inoltre, diversamente a quanto accade per la semplificazione di testi in ambito amministrativo, istituzionale ed educativo destinati a persone multilingue, o a bambini con e senza disabilità (si veda tra gli altri, De Mauro, 2021; Cortelazzo, 2015; Fortis, 2003), i dati e le linee guida sull'adattamento dei testi di informazione in lingua italiana destinati ad adulti con danni cognitivi sono meno diffusi (Piemontese, 1996; Dell'Orletta et al., 2014).
Di contro, molti studi indicano che le procedure di semplificazione linguistica, sia manuale che automatica, sono un valido aiuto per la fruizione di contenuti testuali per diverse categorie di disturbi linguistici e cognitivi (si veda Siddharthan, 2014 per una rassegna).

In questo scenario, il presente lavoro propone l'integrazione di strumenti di analisi linguistica e psicolinguistica come possibile strategia per implementare procedure di semplificazione dei contenuti dei TG al fine di migliorare l'esperienza di fruizione da parte di persone con danno cerebrale acquisito.

2 Metodo

La ricerca è stata articolata in diverse fasi descritte nei paragrafi seguenti.

2.1 Selezione dei testi originali delle notizie

Sono stati scelti 2 TG trasmessi nel mese di settembre del 2020 dalle reti RAI: il TG1 di RAI Uno e il TG di RAI News 24.
Per ciascun TG sono state scelte 3 edizioni:
- edizione lunga serale (lancio delle notizie in versione estesa);
- edizione breve mattutina (lancio delle notizie in versione sintetica);
- versione in LIS (lancio delle notizie tradotto).

Sono state selezionate 18 notizie, 3 per ciascuna edizione di ogni testata. Sono state ottenute notizie su politica (6), Covid-19 (1), economia (1), cronaca (5), spettacolo (1), attualità (3) e sport (1). Nessun argomento oggetto della notizia è stato ripetuto.

2.2 Produzione di versioni semplificate delle notizie

I testi delle notizie sono stati sottoposti a due tipi di rielaborazione (S1e S2) finalizzati alla semplificazione del testo mantenendo inalterati gli aspetti di contenuto dei testi stessi.
La selezione dei parametri da manipolare in fase di semplificazione è stata basata sulle evidenze in ambito neuropsicologico e psicolinguistico sui processi di lettura e comprensione dei testi in persone con disturbi cognitivi acquisiti (Alyahya et al., 2020; Body et al., 1999; Channon & Watts, 2003; Los, 2016; McDonald, 1992; Osterhout & Swinney, 1993; Zurif, & Swinney, 1994; Snow & Douglas, 2017; Turkstra & Politis, 2017).

Parametri manipolati per produrre la versione S1

La procedura di semplificazione per la creazione della versione S1 ha interessato parametri formali, lessicali e sintattici. In particolare, quando possibile:
1. le parole lunghe e quelle a bassa frequenza d'uso sono state sostituite con sinonimi o parole equivalenti più corte e con frequenza più alta;
2. è stato ridotto il numero di proposizioni subordinate;
3. è stato reso esplicito il soggetto;
4. è stato favorito l'uso della costruzione attiva del verbo.

Parametri manipolati per produrre la versione S2

In S2 è stata modificata l'organizzazione degli aspetti contenutistici della notizia e sono state mantenute le semplificazioni lessicali e sintattiche dell'intervento precedente. Lo scopo era di rendere saliente il focus della notizia rispetto alle informazioni marginali. In particolare**:**
1. è stato modificato l'ordine delle informazioni, sia all'interno della frase che all'interno del testo completo delle notizie;
2. è stato ridotto il numero di connessioni implicite tra i diversi elementi della notizia.

Per un esempio di confronto tra versione originale di una notizia e le versioni S1 e S2 si veda la Tabella 1.

Notizia originale tratta dall'edizione LIS del 27/09/2020 del TG di RAI News 24

Testo originale	*A Saluzzo, sua città natale, è stato ricordato il Generale Carlo Alberto Dalla Chiesa nella ricorrenza del centenario della nascita. Per l'occasione, è stato emesso anche un francobollo commemorativo che mostra un ritratto del generale. Presente il Ministro della Difesa Guerini che ha detto: "non abbassò mai lo sguardo".*
S1	*Il Generale Carlo Alberto Dalla Chiesa è stato ricordato a Saluzzo, la sua città natale. Si celebrava il centenario della nascita. Per questa ricorrenza, è stato emesso un francobollo con il suo ritratto. Il Ministro della Difesa Guerini ha partecipato e ha detto: "il Generale non abbassò mai lo sguardo".*
S2	*Sono passati 100 anni dalla nascita del generale Carlo Alberto Dalla Chiesa. Oggi, Saluzzo, sua città di nascita, ha celebrato l'anniversario. Per l'occasione, è stato emesso un francobollo con il ritratto del Generale. Il Ministro della Difesa Guerini era presente. Ha detto: "il Generale non abbassò mai lo sguardo".*

Tabella 1: Esempio di testo di notizia nella versione originale e nelle versioni S1 e S2

2.3 Analisi delle notizie selezionate

I testi originali e semplificati delle notizie selezionate sono stati sottoposti a due procedure di analisi, una manuale-psicolinguistica ed una automatica-linguistica.
L'analisi psicolinguistica condotta manualmente è stata orientata a ottenere una descrizione oggettiva delle principali variabili che influenzano i processi cognitivi alla base dell'elaborazione linguistica: la frequenza d'uso delle parole-contenuto nello scritto (CoLFIS, Bertinetto et al., 2005) e nel parlato (VoLIP, De Mauro et al., 1993), la lunghezza delle frasi, il numero di frasi principali e di frasi subordinate[1] (Tabella 2).

	Originale	S1	S2
Frequenza delle parole-contenuto *(media dei valori nell'italiano scritto)*	1974	2505	2669
Frequenza delle parole-contenuto *(media dei valori nell'italiano parlato)*	244	215	182
Lunghezza delle frasi *(media del numero di parole)*	8	8	8
Numero di frasi principali	4	5	6
Numero di frasi subordinate	3	2	2

Tabella 2: Risultati dell'analisi psicolinguistica manuale

L' analisi linguistica è stata eseguita con strumenti automatici (READ-IT, Dell'Orletta et al., 2011) per ricavare indici specifici legati alla complessità e leggibilità dei testi. In particolare, sono stati ricavati 5 indici: READ-IT base, READ-IT lessicale, READ-IT sintattico, READ-IT globale e GULPEASE (Tabella 3).

	Originale	S1	S2
READ-IT base	36	11	6
READ-IT lessicale	96	80	83
READ-IT sintattico	51	13	7
READ-IT globale	80	34	21
GULPEASE	51	58	65

Tabella 3: Risultati dell'analisi automatica

Per valutare l'efficienza del processo di riformulazione dei testi, le metriche di complessità automatiche e i valori delle valutazioni psicolinguistiche delle versioni originali e semplificate dei testi sono stati sottoposti a *t*-test.
I risultati ottenuti sono i seguenti.

1. Gli indici ricavati dall'analisi psicolinguistica manuale non hanno evidenziato differenze significative tra le tre versioni delle notizie.
2. Al contrario, considerando gli indici dell'analisi linguistica automatica, i testi originali sono risultati più complessi di S1 e di S2 su tutti e 5 gli indici considerati.

Originali vs. S1:

- READ-IT base: t (17) = 3.45, p<.005;
- READ-IT lessicale: t (17) = 2.26, p<.05;
- READ-IT sintattico: t (17) = 3.5, p<.01;
- READ-IT globale: t (17) = 5.82, p<.001; GULPEASE: t (17) = -4.33, p<.001.

Originali vs. S2:

- READ-IT base: t (17) =4 .11, p<.001;
- READ-IT lessicale: t (17) = 2.03, p=.05;
- READ-IT sintattico: t (17) = 4.31, p<.001;
- READ-IT globale: t (17) = 8.46, p<.001; GULPEASE: t (17) = -5.34, p<.001.

I testi S2 sono risultati più semplici dei testi S1 solo per gli indici READ-IT base (t (17) = 215, p<.05) e GULPEASE (t (17) = -5.08, p<.001).

2.1 Verifica sperimentale-pilota basata sulla prestazione di parlanti sani

I testi originali e semplificati delle notizie sono stati usati in 2 esperimenti con parlanti adulti sani: E1 ed E2.
E1 è stato organizzato come esperimento pilota per verificare l'impatto sulla lettura delle semplificazioni S1 e S2 rispetto alla versione originale.
Sono stati registrati i tempi di lettura ed è stata valutata la ritenzione/comprensione della notizia attraverso una domanda di verifica del tipo vero/falso.
E2 aveva lo scopo di ottenere informazioni su ulteriori potenziali fattori rilevanti per i processi di lettura e comprensione dei testi che sfuggono al controllo degli strumenti di analisi manuale e automatica già impiegati. In particolare, in E2 sono stati analizzati tre aspetti dei testi delle notizie usati come stimoli in E1:

1. La complessità dell'argomento della notizia.

È plausibile supporre che aspetti legati al contenuto semantico specifico di un tema economico, sportivo

[1] Il calcolo del numero di frasi principali e subordinate è stato condotto in modo indipendente dalle autrici e supervisionato da un terzo annotatore linguista esperto. È opportuno specificare che per il calcolo del numero di frasi principali e subordinate non è stato possibile sfruttare l'analisi automatica eseguita tramite lo strumento READ-IT (Dell'Orletta et al., 2011). Il motivo risiede nel fatto che nei testi selezionati per questo studio era presente un elevato numero di frasi ellittiche di verbo, caratteristica molto frequente nel linguaggio dei notiziari televisivi, ma che sfugge ai calcoli dello strumento.

o di politica estera richiedano l'uso di termini non semplificabili o, anche, che il tema stesso risulti più o meno complesso per il lettore e influisca, pertanto, sulla rapidità di lettura e/o sulla comprensione della notizia.

2. Il livello di difficoltà della formulazione linguistica della notizia.

È noto che gli strumenti di analisi automatica dei testi forniscono informazioni sulla leggibilità e l'accessibilità della lingua dell'emittente del messaggio. Tuttavia, la difficoltà percepita dal ricevente potrebbe non essere congruente con esse. Ciò potrebbe rivelare elementi utili all'analisi dei processi di lettura e comprensione.

3. La naturalezza della lingua italiana usata nella stesura del testo della notizia.

È possibile che le trasformazioni operate sul testo, benché utili a migliorarne la leggibilità, possano risultare artefatti e innaturali per il lettore e influenzarne lettura e comprensione.

Partecipanti

Hanno partecipato allo studio 54 studenti dell'Università degli Studi di Salerno (età media = 25 anni) di madre lingua italiana[2].

Stimoli

Il set di 54 notizie (18 notizie originali, 18 S1, 18 S2) è stato suddiviso in 3 liste per evitare effetti di ripetizione del materiale. In ciascuna lista, ogni notizia era presente in una sola versione. Ciascuna lista era composta da 6 notizie originali, 6 S1 e 6 S2. Per ciascuna lista sono stati creati 3 diversi ordini di presentazione delle notizie per bilanciare eventuali effetti di affaticabilità e di novità del compito.
I partecipanti sono stati assegnati a ciascuna lista e ordine di presentazione in modo casuale.

E1: Procedura

Il compito di lettura è stato eseguito dai partecipanti in modalità autogestita[3]. Ciascun partecipante ha ricevuto via mail due file implementati in ambiente Microsoft PowerPoint in cui era attivata la possibilità di registrare il tempo trascorso dal partecipante sulle dispositive nelle quali era trascritto il testo delle notizie (Font: Calibri; 28 punti).
Il file 1 aveva la funzione di familiarizzare con la procedura.
Il file 2 era il file sperimentale e veniva ricevuto dal partecipante solo dopo la verifica della corretta esecuzione della fase di familiarizzazione. Ai partecipanti veniva chiesto di leggere ciascuna notizia e di rispondere ad una domanda di verifica con due opzioni di risposta (vero o falso).

E1: Risultati

I dati relativi alla prestazione dei partecipanti che hanno fornito un numero di risposte errate alla domanda di verifica superiore a 1,5 deviazioni standard (DS) rispetto alla media del campione sono stati esclusi dalle analisi statistiche. L'applicazione di questo criterio ha determinato l'esclusione di 5 partecipanti. I tempi di lettura complessivi dei testi, i tempi medi di lettura per parola e il numero di risposte errate prodotte alla domanda di verifica non hanno mostrato differenze statisticamente significative nel confronto tra le tre versioni dei testi delle notizie (Tabella 4).

	tempo complessivo (ms*)	**tempo per parola (ms)**	**errori (%)**
Originale	Media=17044 DS= 9942	Media=328 DS=151	6,7%
S1	Media=17377 DS=10119	Media=343 DS=187	5,6%
S2	Media=17095 DS= 9852	Media=327 DS= 156	4,8%

* ms= millisecondi

Tabella 4: Tempi di lettura ed errori per le tre versioni delle notizie

E2: Procedura

E2 è stato implementato su Google-Module. Almeno 7 giorni dopo aver preso parte ad E1, gli studenti arruolati hanno ricevuto un Module contenente una delle 3 liste di 18 notizie selezionate. La lista ricevuta per E2 era diversa rispetto a quella ricevuta per E1.
Per ogni notizia è stato chiesto ai partecipanti di esprimere un giudizio da 1 a 5 per valutare:
- la Complessità della notizia, ovvero quanto l'argomento della notizia fosse difficile da comprendere;
- la Formulazione linguistica, ovvero quanto la lingua usata per formulare il testo della notizia fosse difficile da comprendere a prescindere dal contenuto;
- la Naturalezza, ovvero quanto la lingua usata per

[2] Nel campione dei partecipanti era presente uno studente bilingue (L1, arabo) residente in Italia dall'infanzia con un'ottima padronanza dell'italiano.

[3] A causa della situazione sanitaria e delle normative vigenti durante la fase di raccolta dei dati, non è stato possibile condurre l'esperimento in laboratorio. I partecipanti hanno ricevuto le istruzioni e l'addestramento alla procedura sperimentale di esecuzione del compito e di registrazione dei risultati nel corso di 3 riunioni virtuali. L'esperimento è stato autogestito dai partecipanti presso le proprie abitazioni e attraverso l'uso delle proprie attrezzature.

formulare la notizia corrispondesse al modo in cui i parlanti italiani usano normalmente la lingua per comunicare in diversi contesti.

E2: Risultati

Diversamente da quanto osservato per l'analisi automatica dei testi, i giudizi dei parlanti non hanno mostrato differenze significative tra le diverse formulazioni delle notizie (Tabella 5).

	Originale	S1	S2
ARGOMENTO	2,2	2,1	2,0
FORMULAZIONE	2,3	2,2	2,2
NATURALEZZA	3,4	3,6	3,5

Tabella 5: Risultati delle valutazioni soggettive dei parlanti

3 Conclusioni

Il presente lavoro ha descritto la messa a punto e l'applicazione preliminare di un metodo basato sull'integrazione di strumenti di analisi linguistica automatica disponibili in lingua italiana con strumenti psicolinguistici e conoscenze neuropsicologiche per la produzione di versioni semplificate dei testi di notiziari televisivi da destinare a persone con disturbi cognitivo-linguistici.

Il metodo usato ha permesso di rispondere a precise esigenze empiriche. I testi dei notiziari hanno lo scopo di divulgare con completezza di informazione i temi più disparati in un formato breve e poco ridondante. Questo pone vincoli alla semplificazione linguistica: non tutte le parole possono essere sostituite, né alcuni particolari di dettaglio possono essere omessi. Inoltre, la complessità concettuale di alcuni temi oggetto delle notizie può essere indipendente dalla formulazione linguistica. Il metodo proposto ha permesso di lavorare su molteplici piani: la manipolazione di parametri linguistici (S1) e di organizzazione del contenuto (S2, valutazioni soggettive della complessità del contenuto in E2); l'analisi della lingua dell'emittente (valutazioni psicolinguistiche manuali e valutazioni linguistiche automatiche) e del ricevente (valutazioni soggettive dei parlanti in E2).

Il confronto tra le misurazioni derivanti dall'analisi psicolinguistica manuale, dalle valutazioni soggettive dei parlanti e dagli indici ricavati dall'analisi linguistica automatica non sono risultati coerenti tra loro. Gli strumenti di analisi automatica si sono rivelati più sensibili nel cogliere la manipolazione della complessità dei testi rispetto alle analisi psicolinguistiche e alle valutazioni soggettive dei parlanti. Questo risultato mette in evidenza l'importanza di integrare i metodi di analisi nella progettazione di interventi di semplificazione.

I dati comportamentali preliminari (tempi di lettura e accuratezza alla domanda di verifica) della sperimentazione-pilota con partecipanti sani hanno mostrato un'assenza di impatto significativo degli indici di analisi automatica relativi alla leggibilità e alla complessità dei testi sulla prestazione di lettura.

Ragioni di ordine metodologico spiegano questi risultati. In primo luogo, occorre considerare che, piuttosto prevedibilmente, la competenza linguistica dei partecipanti arruolati (giovani, sani e ben scolarizzati) ha neutralizzato l'effetto del livello di complessità del materiale presentato. Il dato è, infatti, coerente con uno studio di Crossley et al. (2014) in cui è stato verificato che differenze statisticamente significative nella velocità di lettura di testi di diversa complessità sono annullate dall'inserimento del livello di competenza del lettore come covariata nelle analisi statistiche. D'altra parte, non è da escludere che una replica dello studio con l'uso di un *eye-tracker* possa rivelare aspetti qualitativi e quantitativi più dettagliati sulla fluidità del processo di lettura nelle tre versioni dei testi considerati: ad esempio, durata delle fissazioni su determinate porzioni di testo, fissazioni multiple di porzioni specifiche del testo, parole saltate. Inoltre, la brevità dei testi usati non ha permesso di indagare in profondità aspetti fini legati alla comprensione e ritenzione del materiale in cui è più probabile osservare effetti della semplificazione testuale. La comprensione è stata, di fatto, considerata marginalmente attraverso una sola domanda di verifica. In tal senso, è importante sottolineare che, a parità di livello di comprensione di un testo, il parametro che distingue in maniera significativa la prestazione dei lettori sani rispetto a lettori con disturbi cognitivi è la velocità di lettura (Webster et al., 2018). Questo aspetto sarà valutato in una fase successiva dello studio in cui la procedura sperimentale qui descritta sarà estesa a persone con danni cognitivi conseguenti a lesioni cerebrali. Per ottenere misurazioni più accurate e per creare un contesto di somministrazione più adatto ai partecipanti sarà apportata una modifica al paradigma sperimentale impiegato come compito di lettura. Sarà implementato un compito di lettura con la tecnica delle finestre in movimento (*moving windows*, Witzel et al., 2012). Questo paradigma consente la presentazione del testo parola per parola, o sintagma per sintagma. L'avanzamento da un'unità testuale alla successiva è gestito dal lettore e consente la registrazione dei tempi di lettura per le singole unità considerate.

Riferimenti bibliografici

Alyahya, R. S., Halai, A. D., Conroy, P., & Lambon Ralph, M. A. (2020). A unified model of post-stroke language deficits including discourse production and their neural correlates. *Brain*, *143*(5), 1541-1554.

Bertinetto, P. M., Burani, C., Laudanna, A., Marconi, L., Ratti, D., Rolando, C., & Thornton, A. M. (2005). CoLFIS (Corpus e Lessico di Frequenza dell'Italiano Scritto). *http://www. istc. cnr. it/material/database*.

Body, R., Perkins, M., & McDonald, S. (1999). Pragmatics, cognition, and communication in traumatic brain injury. *Communication disorders following traumatic brain injury*, 81-112.

Cartwright, J., & Elliott, K. A. (2009). Promoting strategic television viewing in the context of progressive language impairment. *Aphasiology*, *23*(2), 266-285.

Channon, S., & Watts, M. (2003). Pragmatic language interpretation after closed head injury: Relationship to executive functioning. *Cognitive Neuropsychiatry*, *8*(4), 243-260.

Cortelazzo, M. A. (2015). La semplificazione dei testi amministrativi. Le buone pratiche. *In: Bombi R. (a cura di), Quale comunicazione tra Stato e cittadino oggi? Per un nuovo manuale di comunicazione istituzionale e internazionale,* Il Calamo, Roma, pp. 93-110.

Crossley, S. A., Yang, H. S., & McNamara, D. S. (2014). What's so Simple about Simplified Texts? A Computational and Psycholinguistic Investigation of Text Comprehension and Text Processing. *Reading in a Foreign Language*, *26*(1), 92-113.

De Mauro T., Mancini F., Vedovelli M., & Voghera, M. (1993) Lessico di frequenza dell'italiano parlato, Milano, Etaslibri.

http://www.parlaritaliano.it/index.php/it/volip

De Mauro, T. (2021). *Guida all'uso delle parole: parlare e scrivere semplice e preciso per capire e farsi capire.* Gius. Laterza & Figli Spa.

Dell'Orletta, F., Montemagni, S., & Venturi, G. (2011, July). Read–it: Assessing readability of Italian texts with a view to text simplification. In *Proceedings of the second workshop on speech and language processing for assistive technologies* (pp. 73-83).

Dell'Orletta, F., Wieling, M., Venturi, G., Cimino, A., & Montemagni, S. (2014, June). Assessing the readability of sentences: which corpora and features? In *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications* (pp. 163-173).

Denicolai, L. (2016). Parole sul video. Linguaggio audiovisivo, comunicazione e afasia. *Parole sul video: linguaggio audiovisivo, comunicazione e afasia*, 318-331.

Di Luca, M., Baker, M., Corradetti, R., Kettenmann, H., Mendlewicz, J., Olesen, J. & Westphal, M. (2011). Consensus document on European brain research. *European Journal of Neuroscience*, *33*(5), 768-818.

Fortis, D. (2003), Il plain language. Quando le istituzioni si fanno capire, *I quaderni del mestiere di scrivere*, 5, 1-24.

Los, M. A. (2016). *Psycholinguistic factors influencing word retrieval in subjects with aphasia*. Rush University.

McDonald, S. (1992). Differential pragmatic language loss after closed head injury: Ability to comprehend conversational implicature. *Applied Psycholinguistics*, *13*(3), 295-312.

Osterhout, L., & Swinney, D. A. (1993). On the temporal course of gap-filling during comprehension of verbal passives. *Journal of Psycholinguistic Research*, *22*(2), 273-286.

Piemontese, M. E. (1996), *Capire e farsi capire. Teorie e tecniche della scrittura controllata*, Tecnodid, Napoli.

Siddharthan, A. (2014). A survey of research on text simplification. *ITL-International Journal of Applied Linguistics*, 165(2), 259-298.

Snow, P.C. & Douglas, J. (2017) Psychosocial aspects of pragmatic language difficulties. *In: Cummings L. (eds) Research in Clinical Pragmatics. Perspectives in Pragmatics, Philosophy & Psychology, vol 11*. Springer, Cham.

Turkstra L.S., Politis A.M. (2017) Traumatic Brain Injury. *In: Cummings L. (eds) Research in Clinical Pragmatics. Perspectives in Pragmatics, Philosophy & Psychology, vol 11*. Springer, Cham.

Webster, J., Morris, J., Howard, D., & Garraffa, M. (2018). Reading for meaning: What influences paragraph understanding in aphasia? *American Journal of Speech-Language Pathology,* 27(1S), 423-437.

Witzel, N., Witzel, J., & Forster, K. (2012). Comparisons of online reading paradigms: Eye tracking, moving-window, and maze. *Journal of Psycholinguistic Research,* 41(2), 105-128.

Zurif, E., & Swinney, D. (1994). The neuropsychology of language. *Handbook of psycholinguistics*, 1055-1074.

Dialogue Analysis with Graph Databases: Characterising Domain Items Usage for Movie Recommendations

Martina Di Bratto, Maria Di Maro, Antonio Origlia, Francesco Cutugno
URBAN/ECO Research Center - University of Naples "Federico II"
martina.dibratto@unina.it, maria.dimaro2@unina.it,
antonio.origlia@unina.it, cutugno@unina.it

Abstract

Nowadays, the use of graph databases combined with textual corpora analysis seems to play a pivotal role in supporting dialogue systems design and implementation. However, dialogues are rarely put in an explicit relationship with the graph structures representing the knowledge domain. In this work, we show how native graph databases provide a framework for a deeper understanding about the use of domain items during dialogue. We describe a multiple-source data collection procedure and we describe how linguistic concepts related to common ground can be found in graph structures. We also describe different patterns that can be detected in the obtained graph structures and discuss their implications in the design of dialogue systems for the movie recommendation task.

1 Introduction

Graph-based data have become popular as support tools for a number of Natural Language Processing tasks, from Word Sense Disambiguation using node embeddings (Yao et al., 2017), to Knowledge Base Collection (Yu et al., 2020) and Fraud Detection (Srivastava and Singh, 2018; Stray, 2019). Most recent approaches involving the use of graph databases converge towards numerical representations of the included items for use by machine learning algorithms (Chanpuriya et al., 2020; Yang et al., 2020b). Knowledge structures using Labelled Property Graphs, however, are typically designed to be more interpretable by human researchers, who can also setup informative queries to extract latent knowledge from a number of cross-referenced resources. This suggests that the same resources supporting dialogue systems can be queried by human experts to extract deeper understanding about the use of domain items, supporting explainability. This implies that it is necessary to explicitly cross-reference dialogue corpora with domain knowledge, while the two are typically considered separately.

In this paper, we describe a multiple-source data collection procedure to cross-reference an annotated English dialogue corpus covering the movie recommendation task, the Internet Movie Database and Wikidata. We present the relationship *common ground* concepts (see Section 3) have with graph structures and we present a set of examples concerning dialogue analysis performed in the graph. The main research questions are:

- Q1: How to represent dialogues and domain knowledge in a single graph structure?
- Q2: Is it possible to formalise common ground concepts guiding dialogue analysis in the resulting graph?
- Q3: Is it possible to use the resulting graph to extract interpretable patterns to guide dialogue systems design?

The paper is organised as follows: in Section 2 we describe the considered resources and how they were cross-referenced in a graph database. In Section 3, we provide a deeper discussion about Common Ground representations and their relationship with graph databases. Section 4 shows how dialogue history, connected with domain knowledge, can be queried to extract knowledge characterising the use of domain items in the considered (sub-)dialogues. This is of interest for the open challenge concerning the study of Argumentation-Based Dialogue which, as opposed to Argumentation-Based Inference, is an area that is still lacking a theoretical framework of reference (Prakken, 2018).

2 Materials and Methods

To demonstrate how a single, graph-based, resource can be built to support corpus-based linguistic analyses, we present, first of all, the assembling procedure of a graph database in the movie recommendation domain. Specifically, we present how freely available information about the movies domain were imported and integrated with data coming from a corpus of chat-based interactions between human subjects.

2.1 The Inspired Corpus

For the purpose of this study, the Inspired corpus is considered (Hayati et al., 2020). It is a recommendation dialogue dataset of two-paired crowd-workers who chat in a natural setting in English. In each conversation, different roles have been assigned to each participant: one acts as the recommender, while the other acts as the movie seeker. The aim of the recommenders is to recommend a movie to the seekers following their preferences, thus achieving the conversational goal successfully. Sociable recommendation strategies are annotated by two experts with a linguistic background, based on past social science studies. The annotation schema is composed by a set of persuasive strategies that has been divided in two categories: preference elicitation strategies and sociable strategies. The whole dataset consists of 1,001 dialogues, with a total of 35,811 utterances, where each recommender's utterance is manually annotated with the corresponding sociable strategies. Domain items like movies, people, genres and plots are tagged in the transcriptions. The dataset presents a average of turns per dialogue equal to 10.73 since recommenders are asked to continue the chat for a minimum of 10 turns.

2.2 Cross-Domain Graph Design

In this work, we adopt Neo4J (Webber, 2012): an open source graph database manager that has been developed over the last 16 years and applied to a high number of tasks related to data representation (Dietze et al., 2016), exploration (Drakopoulos et al., 2015) and visualisation (Jiménez et al., 2016). Neo4j is characterised by high scalability and ease of use. It is a native graph database using data structures that, differently from other graph based approaches, like the ones based on RDF, are designed for performance speed and optimised for graph traversal operations. In general terms, approaches based on RDF are designed for compatibility and general purpose knowledge representation while graph databases are more application oriented. Neo4j has been used in (Sansonetti et al., 2019) as part of a social recommender system based on friends networks extracted from Facebook and on cultural heritage data coming from DBPedia and Europeana. Automatically collecting and organising large amounts of data is relatively easy, nowadays, given the availability of Linked Open Data (LOD). Graph databases, moreover, provide the necessary flexibility to cross-reference different resources, enabling dialogue analysis in relationship with domain knowledge representation. The knowledge base for the movies domain is built by collecting data from different sources and organising it so that cross-referencing is possible. The first step consists in importing the data provided by the Internet Movie Database (IMDB), which is the most structured data source available for the specific domain. The main node labels imported after this step are MOVIE and PERSON. Moreover, for the MOVIE nodes, genres are also represented as labels (e.g. WESTERN, THRILLER, etc...). MOVIE and PERSON nodes are linked by WORKED_IN, WROTE and DIRECTED relationships, mainly. IMDB does not report information about awards won by people and movies. This however, is an important information to consider when ranking potential recommendations. Such information is found in Wikidata, which also contains the IMDB ids for both movies and persons (P345). Connecting Wikidata to IMDB is straightforward, given the already existing alignment. The *award_received* (P166) relationship in Wikidata connects movies and awards while also providing optional qualifiers to further detail the relationship. The qualifier we consider for *award_received* is *winner*, connecting the relationship to one or multiple people. Awards are represented by AWARD nodes and each specific award is represented by a node labelled as AWARDINSTANCE. This way, each AWARDINSTANCE can be connected to multiple winners, to the movie it was awarded for and to the AWARD node it represents.The second part of the procedure covers the import of dialogue data from the Inspired corpus.

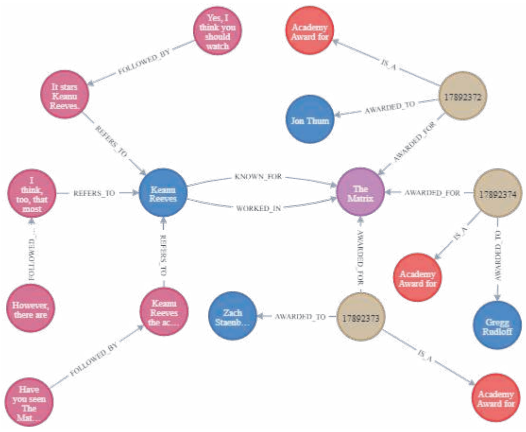


Figure 1: An extract of the final graph structure, including MOVIEs (purple), PERSONs (blue), AWARDINSTANCEs (brown) of AWARDs (red).

Since Inspired is a dialogue corpus constituted by a sequence of turns between two subjects acting either as an information seeker or as a domain expert recommender, the most natural way to represent dialogue history in a graph form is by a nodes chain. Therefore, each turn is represented by an UTTERANCE node also having a RECOMMENDER/SEEKER label representing who stated the sentence. Since there are no returning users in Inspired, there is no need to model them as separate nodes. UTTERANCE nodes expressing specific intents, were marked with labels corresponding to the Inspired sociable strategies. UTTERANCE nodes were linked using FOLLOWED_BY relationships to keep track of the turns sequences. Connecting Inspired to the data collected from the web is performed using text matches between substrings marked as named entities in the original dataset (e.g. movies, people, etc...) and the name fields collected from IMDB and Wikidata. At this time, potential ambiguities are introduced in the graph due to homonyms: these are later solved during graph analysis, as discussed in Section 4. The final structure of the graph is shown in Figure 1. We are not able to share the database, as we cannot redistribute IMDB data. However, the code to generate it, once the data is obtained from the official source, is freely available[1]. The final graph contains approximately 17 million nodes and 75 million relationships.

[1] https://github.com/antori82/MS_MovieGraph

3 Common Ground and Dialogues

Conversing is a joint activity, for which goals and roles of the interlocutors must be identified in order to reach the conversational targets (Macagno and Bigi, 2017). For this purpose, mutual understanding is so fundamental that messages need to be encoded upon a *common ground*, defined as the "[...] presumed background information shared by participants in a conversation" (Stalnaker, 2002, p. 1). Common ground, as (Clark, 2015) acknowledged, can be of four main types: personal, local, communal and specialised. In this work, we focus on personal and communal common ground. *Personal Common Ground* (PCG) is established collecting information over time through communicative exchanges with an interlocutor and it can be considered as a record of shared experiences with that person. This specific set of information can also be considered, as in this work, as part of what builds the *Personal Experience* (PE) of an interlocutor, useful in the future steps of the interaction or for future interactions. *Communal Common Ground* (CCG) refers to the amount of information shared with people belonging to the same community, such as general knowledge, knowledge about social background, education, religion, nationality, and language(s).

In this work, we present how different sets of shared knowledge can be represented in the form of graphs to be analysed in depth. While the proposal draws significant inspiration from older systems based on inference engines, the aim is to re-interpret the approach using modern technologies that naturally blend with machine learning. Graph-based representations have gained popularity, in recent years, due to their flexibility and expressiveness. In (Yang et al., 2020a), knowledge graphs were used for intent identification in multi-turn dialogues. (Lei et al., 2020) proposes a graph based framework to detect user preferences in multi-turn dialogues. This approach covers Knowledge Graph navigation but it does not represent dialogues in the graph. Differently, (Wang et al., 2021) proposes the use of graph structures both to represent interactions and learn intents automatically. User-item interactions were part of the graph but they were treated as isolated items. We include sequence dynamics as the recorded order from Inspired is explicitly represented.

Graph structures obviously open new perspectives in automatic dialogue management and it is, therefore, important to investigate how linguistic concepts can be used to describe with deeper details the recurring structures forming between dialogues (PE) and domain knowledge (CCG). While the CCG contains entities and relationships between entities that belong to the common knowledge of a particular domain (i.e., in the movie domain: ‘directors direct movies’, ‘George Lucas directed Star Wars’, etc.), the PCG is built considering the CCG to select specific entities during the interaction as being related to one another according to particular relationships. The PCG records the interaction flow with a specific user to also extract possible personalised strategies for future interactions with the same user (i.e., ‘The user X is looking for Sci-fi movies; the user X decides to watch The Matrix’). This sub-graph has a temporary status as, at the end of the interaction, it is incorporated in the PE, which contains information about entities and their relationships extracted from various interactions (i.e., ‘Users who usually look for Sci-fi movies also choose to watch The Matrix’). The PE sub-graph, therefore, is complementary to the information contained in the CCG (i.e., the fact that The Matrix is a Sci-fi movie) for other interactions or different interaction phases, such as the *persuasion dialogue* phase, used to “resolve a conflict of opinion between real agents, who can ask for and provide substantive reasons for their claims” (Prakken, 2018). In the graph structure, we now identify the different types of CG as sub-graph structures, supporting a linguistically oriented view of the obtained resource based on mathematical notations. First of all, we identify the CCG in the set of data that are obtained by importing commonly available resources so that

$$CCG = < V_{lod}, E_{lod} > \tag{1}$$

where V_{lod} is the set of nodes imported from LOD and E_{lod} are the reported relationships among these nodes. PE refers to recorded interactions between RECOMMENDERs and SEEKERs, so

$$PE = < V_d, E_d > \tag{2}$$

where V_d is the set of nodes representing utterances from the Inspired corpus and E_d is the set of chain relationships recording the dialogue evolution. PE contains information that is not accessible by other interlocutors, as it concerns the owning speaker, who can disclose parts of it and provide unverifiable information that other involved actors should trust. The PCG is a subset of the PE involving the dialogue chain representing the ongoing interaction so that $PCG \subseteq PE$. Named entities in the PE graph are linked to items in the CCG by a ϕ function representing the result of the *grounding* process. The domain of the ϕ function D_ϕ consists of utterances containing named entities, while its range R_ϕ consists of CCG items. Links defined by ϕ generate a *reference graph* R defined as

$$R = < V_\phi, E_\phi > \tag{3}$$

where $V_\phi = \{D_\phi \cup R_\phi\}$ and $E_\phi = \{(v_d, v_{lod}) | \phi(v_d) = v_{lod}\}$ where $v_d \in V_d$ and $v_{lod} \in V_{lod}$. The reference graph is represented in Neo4j using REFERS_TO relationships between UTTERANCE nodes and nodes representing named items, like MOVIEs and PERSONs.

4 Patterns in Common Ground

Using graphs to analyse PE and CCG together allows to extract interpretable knowledge to obtain deeper understanding of dialogue phenomena and support an better informed design of dialogue systems with respect to solutions relying on machine learning only. The examples we provide concern items characterisation in relationship with dialogue strategies. Since the presented graph structure captures dialogue dynamics in the form of FOLLOWED_BY relationship chains, we use it to analyse dialogue dynamics and to characterise items selected for the employed strategies w.r.t. previous exchanges. This can put in relationship the PE and the CCG explicitly in the form of ring-like patterns crossing both the PE and the CCG. Formally, let’s consider a path $p = [u_n, c_1, \ldots, c_w, u_{n-k}, \ldots, u_n]$ where $u_i \in V_d$, $c_j \in V_{lod}$, $(u_n, c_1) \in E_\phi$, $(u_{n-k}, c_w) \in E_\phi$ and $\forall\, 0 \leq m < k, (u_{n-m-1}, u_{n-m}) \in E_d$. The pattern can be used both for disambiguation purposes and to extract the context that led interlocutors to discuss a certain item. In the case shown in Figure 2, first the MOVIE *Knives Out* was named and, after a few turns, a person named *Chris Evans*. There are 103 people, in the database, named *Chris Evans* but only one of them can be directly related to the previous subject, *Knives Out*, removing the ambiguity. The pattern involving the naming of a MOVIE followed by a PERSON can be further studied considering the annotated sociable strategies for the RECOMMENDER.

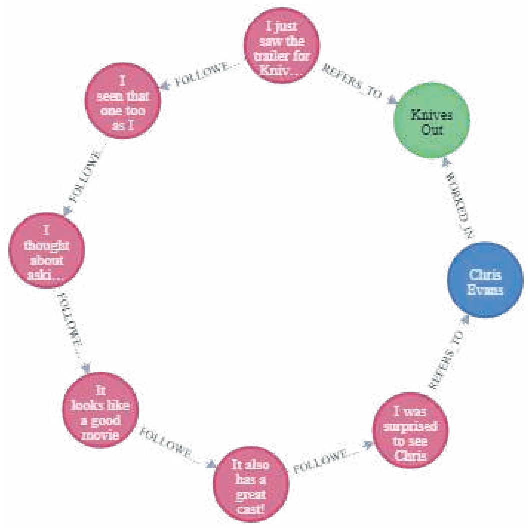


Figure 2: A ring-like pattern found in one the Inspired dialogues.

Let's consider the case of ring-like patterns involving a PERSON who can be related to a previously named MOVIE by a WORKED_IN relationship. In the database, there are 122 patterns matching these constraints. In this case, MOVIEs involved in this pattern are used to support mainly sociable strategies such as *experience inquiry* but also *encouragement* to watch the previously proposed movie. PERSON items, instead, seem to support mostly *credibility*, used by the RECOMMENDER to show expertise and trustworthiness in providing information. The opposite pattern, where PERSONs are named before the MOVIE they WORKED_IN, is less frequent, with 66 occurrences, but presents more variability in the involved strategies. PERSONs support, in this case, *credibility*, *experience inquiry* and *personal opinion* strategies mostly, while MOVIEs support *personal opinion* and *encouragement* strategies. Figure 3 shows a summary of the observed distributions over the considered patterns. We interpreted these data as an indication of the transition between two different recommendation dialogue phases. In particular, PERSON items are mainly used, by RECOMMENDERs, to persuade the SEEKER by showing competence about MOVIEs mentioned during the Exploration phase. Here, strategies like *experience inquiry* are more frequent given that information about SEEKERs' tastes and experiences need to be collected (Gao et al., 2021). An interesting observation arises from the analysis of the inverse pattern as it seems to present a continuity with the previous one. Also here PERSON items are mainly used to support *credibility*, introducing the Exploitation phase, whose aim is to take advantage of the collected information to recommend a potentially valuable item. Indeed, the following MOVIE items involved in the pattern seem to occur in strategies typical of this phase such as *personal opinion* or *encouragement*, mainly used by the RECOMMENDERs to persuade the SEEKER to watch a proposed MOVIE. The pattern that seems to emerge suggests that, during the Exploration Phase, RECOMMENDERs build a model of the SEEKER by collecting previously watched MOVIEs and talking about potentially interesting ones. They show competence about MOVIEs by referencing PERSONs involved in them. The dialogue then blends into the Exploitation phase, where persuasion strategies are adopted.

To further detail the choice of actors involved in dialogues, we consider the relationship actors have with movie genres. Dialogue analysis alone cannot go beyond frequency observations for specific tokens: the cross-referenced database allows to relate named items to the categories items of interest fall into. Specifically, for each PERSON who WORKED_IN a MOVIE, we consider the skewness of the corresponding GENRE distribution. Highly skewed distributions are obtained for PERSONs specialised in a specific GENRE while low skewed distributions are obtained for more *versatile* PERSONs. Interestingly, people with mid-skewed distributions are used more often both w.r.t. specialised PERSONs and w.r.t versatile ones (Figure 4). This suggests that people who work in hybrid genres (e.g. DRAMA/COMEDY) are more *informative* items to discuss. One possible explanation would be that these items provide indications about genre *nuances* or sub-genres.

5 Discussion and Conclusions

Graph databases have a great potential as linguistic tools for dialogue analysis, modelling interaction dynamics and domain knowledge in the same resource. While this representation provides a powerful support to dialogue systems through machine learning, it can be integrated by more informed design based on a qualitative analysis of the same graph structures. We presented a data collection procedure for a large graph database representing dialogue dynamics explicitly linked

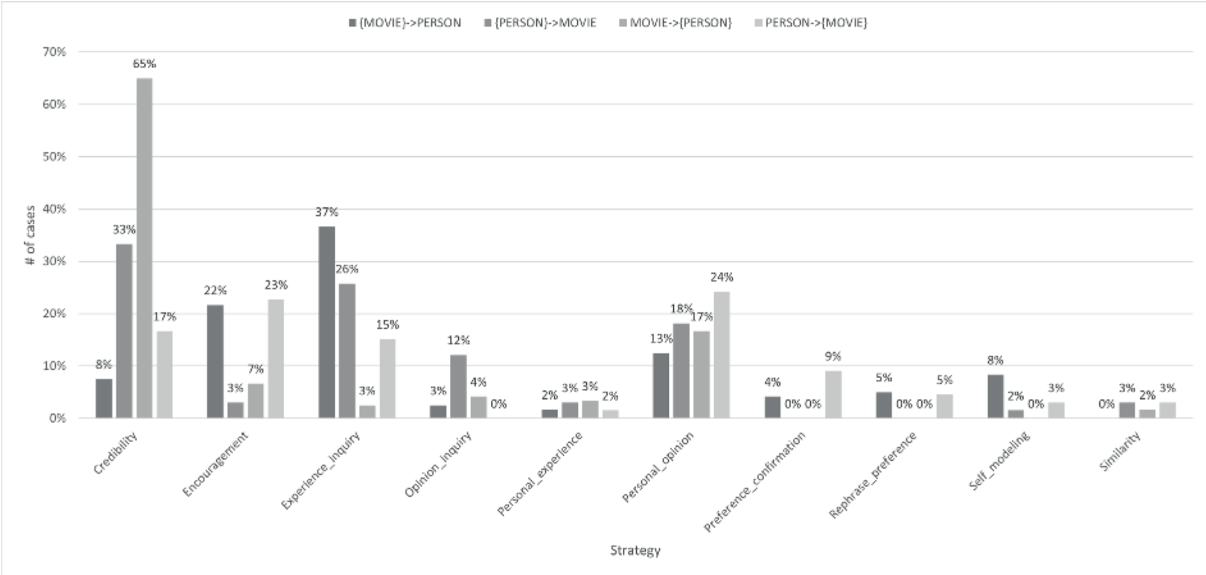


Figure 3: Strategies distribution in ring-like patterns for UTTERANCEs using MOVIE and PERSON items. For each pattern, the item and its position in the considered pattern are found between brackets.

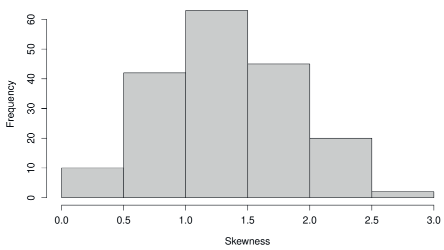


Figure 4: Skewness values distribution.

to domain knowledge. In the representation of such data, different kinds of Common Ground were identified as sub-graphs. We then, used it to investigate the choice of domain items supporting sociable conversational strategies. Concerning the research questions we were interested in investigating, the data collection procedure presented in Section 2.2 presents how a graph database can organise and connect, at the same time, both dialogue history and domain knowledge, providing an example answering Q1 for the specific case of the movies recommendation dialogues. This is, then, generalised in Section 3 by identifying linguistic concepts related to common ground that are described in terms of sub-graphs, independently of the specific domain. Recurring patterns involving both dialogue and common ground structures can, then, be identified using graph traversal queries. Specifically, we have characterised the different use of MOVIE and PERSON nodes depending on the order of appearance in a pattern where the referring nodes are linked by the relationship WORKED_IN, in the CCG. By analysing the sociable strategies involved in the patterns in which the specific domain items may occur, we described how the different use of the same node categories in opposite patterns (i.e., MOVIE followed by PERSON; PERSON followed by MOVIE) can be typical of a specific recommendation dialogue phase. The obtained data describe a way of managing Exploration and Exploitation phases informing the design of conversational recommender systems. We have also shown that, in selecting which PERSON items to use when adopting sociable strategies, RECOMMENDERs seem to prefer PERSONs with mid-skewed distributions over movie genres, suggesting that opinions about these PERSONs may give insight about sub-genres. Future work will consist of empowering the analyses by relating more complex numerical item representations, like node embeddings, to dialogue dynamics.

Acknowledgments

The authors would like to thank Sabrina Mennella for her support in correcting and analysing the Inspired dataset. Maria Di Maro's work is supported by the Italian PON I&C 2014-2020 within the BRILLO project, no. F/190066/01-02/X44.

References

Sudhanshu Chanpuriya, Cameron Musco, Konstantinos Sotiropoulos, and Charalampos Tsourakakis. 2020. Node embeddings and exact low-rank representations of complex networks. *Advances in Neural Information Processing Systems*, 33.

Eve V. Clark. 2015. Common ground. In *The Handbook of Language Emergence*, page 328–353. Wiley, Chichester, UK.

Felix Dietze, Johannes Karoff, André Calero Valdez, Martina Ziefle, Christoph Greven, and Ulrik Schroeder. 2016. An open-source object-graph-mapping framework for neo4j and scala: Renesca. In *International Conference on Availability, Reliability, and Security*, pages 204–218. Springer.

Georgios Drakopoulos, Andreas Kanavos, Christos Makris, and Vasileios Megalooikonomou. 2015. On converting community detection algorithms for fuzzy graphs in neo4j. In *Proceedings of the 5th International Workshop on Combinations of Intelligent Methods and Applications, CIMA*.

Chongming Gao, Wenqiang Lei, Xiangnan He, Maarten de Rijke, and Tat-Seng Chua. 2021. Advances and challenges in conversational recommender systems: A survey. *arXiv preprint arXiv:2101.09459*.

Shirley Anugrah Hayati, Dongyeop Kang, Qingxiaoyang Zhu, Weiyan Shi, and Zhou Yu. 2020. Inspired: Toward sociable recommendation dialog systems. *arXiv preprint arXiv:2009.14306*.

Pablo Jiménez, Javier Villalba Diez, and Joaquin Ordieres-Mere. 2016. Hoshin kanri visualization with neo4j. empowering leaders to operationalize lean structural networks. *Procedia CIRP*, 55:284–289.

Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. 2020. Interactive path reasoning on graph for conversational recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2073–2083.

Fabrizio Macagno and Sarah Bigi. 2017. Analyzing the pragmatic structure of dialogues. *Discourse Studies*, 19(2):148–168.

Henry Prakken. 2018. *Historical overview of formal argumentation*, volume 1. College Publications.

Giuseppe Sansonetti, Fabio Gasparetti, Alessandro Micarelli, Federica Cena, and Cristina Gena. 2019. Enhancing cultural recommendations through social and linked open data. *User Modeling and User-Adapted Interaction*, 29(1):121–159.

Sakshi Srivastava and Anil Kumar Singh. 2018. Graph based analysis of panama papers. In *2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, pages 822–827. IEEE.

Robert Stalnaker. 2002. Common ground. *Linguistics and philosophy*, 25(5/6):701–721.

Jonathan Stray. 2019. Making artificial intelligence work for investigative journalism. *Digital Journalism*, 7(8):1076–1097.

Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. 2021. Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the Web Conference 2021*, pages 878–887.

Jim Webber. 2012. A programmatic introduction to neo4j. In *Proceedings of the 3rd annual conference on Systems, programming, and applications: software for humanity*, pages 217–218. ACM.

Kai Yang, Xinyu Kong, Yafang Wang, Jie Zhang, and Gerard De Melo. 2020a. Reinforcement learning over knowledge graphs for explainable dialogue intent mining. *IEEE Access*, 8:85348–85358.

Luwei Yang, Zhibo Xiao, Wen Jiang, Yi Wei, Yi Hu, and Hao Wang. 2020b. Dynamic heterogeneous graph embedding using hierarchical attentions. *Advances in Information Retrieval*, 12036:425.

Liang Yao, Yin Zhang, Baogang Wei, Zhe Jin, Rui Zhang, Yangyang Zhang, and Qinfei Chen. 2017. Incorporating knowledge graph embeddings into topic modeling. In *Thirty-first AAAI conference on artificial intelligence*.

Haoze Yu, Haisheng Li, Dianhui Mao, and Qiang Cai. 2020. A domain knowledge graph construction method based on wikipedia. *Journal of Information Science*, page 0165551520932510.

Do You Have any Recommendation? An Annotation System for the Seekers' Strategies in Recommendation Dialogues

Martina Di Bratto[1], **Riccardo Orrico**[1], **Ancuta Budeanu**[2], **Marta Maffia**[2] **and Loredana Schettino**[3]
1. University of Naples "Federico II", Italy
2. University of Naples "L'Orientale", Italy
3. University of Salerno, Italy
martina.dibratto@unina.it, mmaffia@unior.it, lschettino@unisa.it

Abstract

The development of dialogue systems benefits from the study of the communication strategies used by human speakers. In the context of recommendation dialogue systems some researchers have investigated the sociable recommendation strategies employed by the Recommenders in natural settings to make successful and persuasive recommendations (Hayati et al., 2020, INSPIRED corpus). However, the Seeker's contribution, as well as the Recommender's, shapes the development of the communicative exchange, in that the Seekers may use specific strategies to disclose their preferences and reach their goal. So, modelling the Seeker's communicative strategies along with the ones used by the Recommender may improve the efficiency of recommendation dialogue systems. In this work, we provide a reliable tagset for the Seekers utterances present in the Inspired dataset, defining a set of communicative strategies coherent with the already existing one for the Recommenders.

1 Introduction

Nowadays conversational recommendation systems seem to be acquiring a fundamental role in information seeking and retrieval. In a recent paper, Hayati and her colleagues (Hayati et al., 2020) have argued for the need to study the communication strategies used by human speakers in a natural setting for developing dialogue systems that are able to make successful and persuasive recommendations. The authors have proposed Inspired, a dataset of recommendation dialogues collected in a realistic setting, enriched with a detailed annotation of the sociable recommendation strategies employed by the Recommender.

However, as in any interaction, these dialogues are the result of the cooperation between the interlocutors, who actively partake in both the construction of meaning and of the relationship among each other (Bazzanella, 2005): the Seeker's contribution, as well as the Recommender's, shapes the development of the communicative exchange, in that the seekers may use specific strategies to disclose their preferences and reach their goal, i.e., to get items that suit their needs. Hence, modelling the Seeker's communicative strategies along with the ones used by the Recommender may improve the efficiency of recommendation dialogue systems.

In this work, we aim to fill this gap proposing a tagset for the Seekers communicative strategies that is coherent with the one previously provided for the Recommenders by Hayati and colleagues. The paper is structured as follows: recommendation dialogue systems are considered in relation to the Argumentation Theory (§ 2) and the Inspired tagset (Hayati et al., 2020) is described (§ 2.1), then the tagset for the Seeker's strategies is presented (§ 3), along with the data proving the reliability of the annotation scheme (§ 3.1) and a preliminary analysis of the interactions (§ 4).[1]

2 Recommendation Dialogue

Recommendation dialogues are characterized by two or more participants who disclose their preference and make recommendation in order to select a certain item that should satisfy the re-

[1] The present study is the result of a collaborative work of all the authors. Paragraphs 2 and 2.1 have been written by Martina Di Bratto, paragraph 3 by Marta Maffia and Ancuta Budeanu, 3.1 and 4 by Riccardo Orrico and, finally, sections 1 and 5 by Loredana Schettino.

quirements retrieved during the communicative exchange. Conversational Recommendation Systems (CoRS), in the same way, aim at finding or recommending the most relevant information (e.g., web pages, answers, movies, products) for users based on textual- or spoken-dialogues, through which users can communicate with the system more efficiently using natural language conversations (Fu et al., 2020). CoRS, thus, can be seen as persuasive social actors since a recommendation can be considered persuasive when it attempts to change people's mind or behavior by employing various persuasive strategies (Shi et al., 2020). A conversation where two or more interlocutors (humans or not) aim to resolve a conflict of opinion, can be considered as a form of persuasion dialogue leveraging on argumentation (i.e., the process of exchanging ideas in order to establish the truth of a statement). CoRSs can be framed in the field of formal argumentation and more specifically, refer to the argumentation-based dialogue. It considers the problems arising from dialogues involving different agents and whose information are shared and distributed among them. This interaction introduces multiple, not necessarily aligned knowledge and, possibly, conflicting goals in the pursuit of a solution to a problem. (Di Maro, 2021). Walton's classification of dialogues (Walton, 1984) is often employed in the study of the argumentation-based dialogue. He distinguished six different categories of dialogue: *persuasion*, *negotiation*, *information seeking*, *deliberation*, *inquiry*, and *quarrel*. The purpose of persuasion dialogues, thus, can be seen as 'pure' argumentation and can be often embedded in other dialogue types (Prakken, 2018). The Recommendation task, indeed, tends to present a pattern structured in two phases, Exploration and Exploitation (E&E), which can be intended as two types of dialogues embedded into each other. According to (Gao et al., 2021, p. 15), with exploration "[...] the system takes some risks to collect information about unknown options". On the other hand, during the exploitation phase, "[...] the system takes advantage of the best option that is known". Hence, the exploration phase can be associated to the *inquiry* dialogue since the main aim is to achieve the "growth of knowledge and agreement" starting from an initial situation of "general ignorance" (Walton and Krabbe, 1995, p. 66). The exploitation phase, on the other hand, starts when the Recommender considers the collected information sufficient to move to the phase whose aim is to resolve a conflict of opinion, i.e. *persuasion dialogue*. During the entire conversation, even if the two participants have a distinct role, they seem to actively interact with each other in order to construct the dialogue meaning and achieve the communicative goal. The Recommender, in fact, is seen as a domain expert who participates actively, guiding the conversation throughout the two phases. The Seekers, who do not have a wide domain knowledge, mostly follow the Recommenders' moves during the exploration phase, while in the exploitation phase they provide implicit or explicit feedback that may lead the Recommender to model the dialogue, eventually finding the most suitable recommendation. Indeed, detecting seekers' communicative intentions is a pivotal process to train a conversational recommender system given that Intent Recognition is responsible for understanding the action that the user is requesting (Iovine et al., 2019). Nonetheless, in a recent review of existing approaches to conversational recommendation (Jannach et al., 2021), the author take note of a still scarce effort in investigating and defining relevant user intents, with a few exceptions considering either domain-independent intents (Cai and Chen, 2019; Narducci et al., 2018, a.o.) or restricted specific subsets (Nguyen and Ricci, 2018, e.g.).

2.1 The Inspired Corpus

The Inspired corpus (Hayati et al., 2020)[2] is a recommendation dialogue dataset of two-paired crowd-workers who chat in a natural setting in English. In each conversation, one participant acts as the Recommender, while the other as the movie Seeker. The aim of the Recommenders is to recommend a movie to the Seekers following their preferences and, thus, achieving the conversational goal successfully. The whole dataset consists of 1,001 dialogues where just the Recommender's utterances are manually annotated with the corresponding strategies. The annotation scheme of the Recommender's utterances is composed by a set of persuasive strategies divided in two categories: preference elicitation strategies and sociable strategies.

Also the collected conversations present the two-phase pattern typical of the recommendation

[2] Dataset and code are freely available online.

task. In the exploration phase preference elicitation strategies are used by the Recommender in order to collect sufficient information regarding the seeker's preferences and tastes about the movie domain. They are divided in *experience inquiry* and *opinion inquiry*.

In the exploitation phase, on the other hand, eight different strategies have been recognized. During this phase, thus, the Recommenders can start the interaction by *offering help* to find the recommendation. They can also express their *personal opinion* or *personal experience* in order to convince the Seekers basing the recommendation on their own experience. Moreover, they can opt for other persuasive strategies such as *credibility*, *similarity*, *encouragement*, *preference confirmation* or *self-modeling* which are mainly used to built rapport with the Seekers, also establishing and improving their role as domain experts.

3 Seeker Annotation

Taking into account the Recommender's annotation scheme proposed by Hayati and colleagues (Hayati et al., 2020) and after an inspection of the dialogues included in the Inspired Corpus, an annotation scheme for Seeker's utterances was developed. The established categories, while covering the domain-specific user intents, are in line with some of the relevant domain-independent ones found in the literature (Jannach et al., 2021, 105), e.g., *Initiate Conversation*, to "start a dialogue with the system"; *Chit-chat*, for "utterances unrelated to the recommendation goal" ; *Provide Preferences*, to "share preferences with the system"; *Ask for Recommendation*, to "obtain system suggestions"; *Obtain Explanation*, to "learn more about why something was recommended"; *Feedback on Recommendation*, to "give feedback on the provided recommendation(s)"; *Quit*, to "terminate the conversation".

We divided Seekers' strategies into four categories.[3]. The first category corresponds to a single strategy, labeled as **recommendation_request** and used by the Seeker to generically ask for a candidate item: *ex. Do you have any recommendations?*

[3]In this pilot stage of the research, we decided to work on the labelling of communicative strategies used by the Seekers in the above mentioned "user information gathering" and "movie recommendation" phases of dialogues. Other strategies, located at the beginning (*greetings*) and at the end of the dialogues (*intentionality, acceptance, refusal*) were also identified but they will not be discussed in this paper

The second category (henceforth called *get_movie*) includes **global requesting strategies**, by which the Seeker can direct the recommendation process on the basis of specific attributes of the movies. They are divided as follows:

- *get_from_genre*, used to ask for a candidate item according to its genre; *ex. What kind of comedy movies do you have to recommend?*
- *get_by_actor*, used to ask for a candidate item featuring a specific actor/actress; *ex. Do you have another movie with Tom Hanks?*
- *get_similar_to*, used to ask for a candidate item with analogous attributes to another specified item; *I would love to see a remake or something similar to Notting Hill.*
- *get_by_year*, used to ask for a candidate item according to its release date; *Do you know anything more recent?*

The third category corresponds to the **giving preference strategies** usually uttered by the Seeker to reply to the Recommender's inquiries:

- *personal_opinion* used to specify personal preferences over candidate items or one/some of their attributes. Also, it can express a positive or negative value towards them; *ex. I liked the acting and the movie itself; I didn't like that movie.*
- *personal_experience*, used to tell about experiences that could be present or not in the past, thus defining if the Seeker have or have not watched that movie; *ex. I saw the trailer for For v Ferrari; No, I haven't seen it.*

Finally, the *get_info* category includes **local requesting strategies** uttered by the Seeker to require information about a specific, recommended movie. This category includes:

- *get_genre*, used to asks about the value of the attribute "genre" for a specified item; *ex. Is it an action movie?*
- *get_acted_in*, used to ask about the movie's cast; *Do you know who else is in the cast?*
- *get_score*, used to request information about the quality evaluation of the movie; *ex. How about the new Rambo?*

- *get_plot*, used to ask about the storyline of a movie; *ex. Could you tell me what the general plot is?*

In order to test the validity of the annotation system, we proceeded to annotate Seekers' utterances taken from the first 20 dialogues between Recommenders and Seekers (331 utterances produced by Seekers) which were annotated by 5 annotators (the authors of this contribution). Each Seeker's utterance could be given one or two labels: a second label was added in those cases in which two strategies were expressed by the Seeker in the same utterance. In most of these cases the assignment of a first and a second label was facilitated by the sequentiality of information in the utterance (ex.the utterance *I recently watched John Wick 3, very good movie, in my opinion and fully action packed* was given *personal_experience* as first label and *personal_opinion* as second label by all the annotators); on the contrary, other cases could present a higher level of ambiguity (for example, in case an annotator intended the utterance *i like the sci-fi movies* to express both *personal_opinion* and *get_from_genre*. In these cases, there was not a unique criterion to identify which one was the first and which one was the second label). Data about annotators agreement and preliminary results of Seekers' strategies based on our annotation are presented in the following sections.

3.1 Annotation Quality

Since the annotation system accounts for the possibility of having two different strategies within the same utterance, the agreement among the 5 annotators could have 3 possible outcomes: for each utterance there could be i) agreement (A), all 5 annotators agreed on both first and second label (type and presence); ii) partial agreement (PA), at least one annotator disagreed on one strategy, though all 5 agreed on the other (e.g. all annotators agree on the first label, but no agreement is reached on the second); iii) disagreement (D), at least one disagreement for both labels.

In most cases (about 85%) the annotators agreed on at least one of the strategies detected. More specifically, A was reached in about 35% of the utterances, while PA in 50% of the cases. D was registered only for 15% of the utterances. The confusion matrix reported below shows more detailed information about the single strategies. Data reported in the matrix are mean percentages of values of the 10 pairs of annotators: label-by-label agreement was first calculated for each pair of annotators and then mean values for all the pairs were extracted and plotted in the matrix to check for which strategies reported, on average, the highest levels of agreement or disagreement across the annotators (Figure 1).

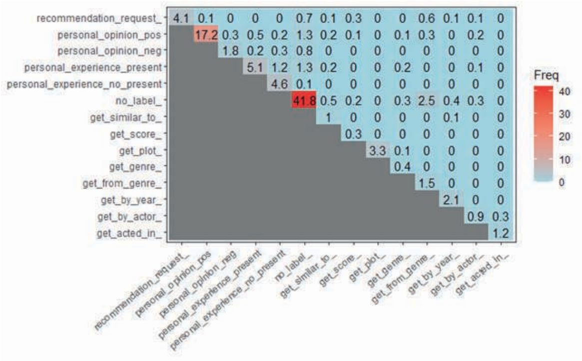


Figure 1: Confusion matrix of the labels assigned by the 5 annotators. The data reported were calculated as mean percentages over the whole dataset of pairwise annotator agreement. No-label refers to the absence of a second strategy; opinion and experience were split into two different strategies according to their evaluation (positive vs negative for experience and present vs not present for experience)

It is clear from the matrix that most cases of disagreement refer to *get_from_genre*. More generally, the matrix shows that among the cases of disagreement, the annotators failed to agree on the assignment of labels relative to global and local requesting strategies, which were often annotated as not representing a specific strategy at all. A sounder measure for the agreement (Fleiss' Kappa) was calculated for those utterance in which all annotators agreed to assign only one label, which amount to about 1/3 of the total of the utterances[4]. The Fleiss' Kappa value obtained for these annotation is 0.887, indicating an overall high agreement among the 5 annotators. The inspection of the score obtained for each specific label shows that while all strategies were detected with a high level of agreement, low values are registered for the category *get_from_genre* (Kappa = 0.247)

[4]The measure was not calculated for the whole data set because of the absence of a stable criterion for ordering strategies in case two were present (see section 3).

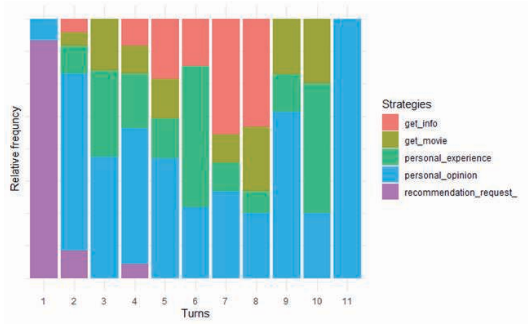


Figure 2: Distribution of the seeker's strategies in each turn of the dialogues.

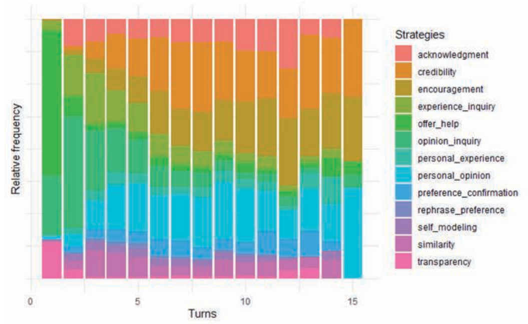


Figure 3: Distribution of the recommender's strategies in each turn of the dialogues. (Hayati et al., 2020)

4 Retrieved Data

This section presents a description of the strategies employed by Seekers in the subset that we analyzed. The data reported here refer to those utterances in which all 5 annotators agreed on the type of strategy detected.

The most frequent strategy is the expression of personal opinions, which alone accounts for almost 50% of the total of the strategies. Of these, the great majority (around 90%) is represented by the strategy 'personal_opinion_pos'. The strategy 'personal_experience' is also quite frequent, amounting to around 20% of the strategies; among these, the expression of absence of experience (*ex. No, I haven't seen that movie*) is more frequent, accounting for more that 60%. Recommendation requests account for 10% of the strategies, while the remainder is made up of those strategies aiming at either collecting specific information about a movie (i.e., get_info) or eliciting a title given a specific preference (i.e., get_movie). Of the former set of strategies, the information that is more frequently asked concerns the plot of the movie, while for the latter, Seekers appear to be most interested in the release date. Although annotated data about the Seekers' turns are referred to a small subset of the whole corpus, it is possible to draw some preliminary strategies on the co-construction of the dialogue by the two participants, by considering the by-turn distribution of the strategies in both participants. As for the Seeker, the different strategies employed are not evenly distributed across the dialogue turns, as shown in Fig.2. The plot shows that recommendation requests are almost the only strategy employed at the beginning of the dialogue, after which their occurrence drops dramatically. On the contrary, the occurrence of get_info and get_movie increase as the dialogue unfolds. Personal opinion and experience, on the other hand, are more evenly distributed, with a drop of their occurrence in the median turns. As for the Recommender, the by-turn distribution of strategies is shown in Fig.3.

The plot shows that, on the Recommender side, the use of the strategy offer_help mirrors the use by the Seeker of a request for recommendation, being employed almost exclusively in the first turn. More generally, the first part of the dialogue is characterized by inquiries, by the Recommender to the Seeker, about his/her opinions and experiences. While the use of these strategies decreases as the dialogue unfolds, strategies aimed at overcoming conflicts (e.g. preference confirmation) or persuading/informing (e.g. encouragement or similarity) are more frequent in the second half of the dialogue. This is mirrored, on the Seeker side, by the use of strategies linked to personal opinions/experiences and global and local requesting strategies.

5 Discussion and Conclusions

This work is supported by the idea that studying communication strategies used by human speakers is fundamental to improve the performances of dialogue systems. This was already supported by Hayati and her colleagues (2020) who analyzed the Recommenders' sociable strategies in recommendation dialogues to develop successful and persuasive recommendation dialogue systems. However, considering the cooperative nature of dialogues, we argue that annotating the Seeker's move may be pivotal in the training phase of recommendation dialogue systems. Hence, we propose an annotation scheme for the Seeker's utterances that is coherent with the annotation of Recommender's utterances. Considering the Seeker's role and main moves, we have drawn four categories: *recommendation requests*, *global requesting strategies*, *giving preference strategies* and *local requesting strategies*. Results on the reliability of the annotation scheme show that the agreement between the 5 annotators ranges from substantial to almost perfect (Landis and Koch, 1977) for most strategies but one, i.e., the strategy used to ask for movies of a specific genre. Similarly, observing the other cases of disagreement, we find that they mostly concern the identification of global and local requesting strategies. We showed that in most of these cases annotators failed to agree on whether an utterance contained a second strategy (manly a specific title request). In this cases, some annotators assigned a second label believing that the more specific request was generated as a conversational implicature stemming from the Seeker's mention of a certain movie title or attribute and the expression of his/her own opinions and experiences. The fact that most of the cases of disagreement fall within this situation might also explain why we registered high levels of disagreement for the *get_from_genre* label. Observing the confusion matrix (Figure 1), what can be noticed is that this category has been frequently confused with the no_label one. An explanation of this phenomenon could be found in utterances like "I love sci-fi movie" to which only the first label as personal_opinion_pos has been assigned. Nonetheless, other annotators also added get_from_genre as second label, for the reason explained above. We believe that this does not specifically depend on the strategy per se, but simply on the fact that genre is the feature of a movie that most frequently was mentioned by the Seekers (30% of the total features, as opposed to i.e. actors and directors, occurring respectively, in 20% and 4% of the cases), therefore more frequently led the annotators to assign different strategies. A finer analysis of the turn by turn strategies of the two participants on a larger number of dialogues would be informative about the extent to which Recomemenders make the inference (and act on it). This would help understand how to treat these cases.

Concerning the general distribution of the Seekers' strategies, positive personal opinion and non-present personal experience seem to be more frequent than the global and local requesting strategies. The strategies distribution along with the dialogue turns, on the other hand, shows that the first turns are mainly characterized by the occurrence of recommendation requests, reflecting the Recommender's strategy of offering help. In the middle of the conversation, requests for getting information or movie titles increase together with personal opinion and personal experience, even if the latter seems to be more equally distributed. This distribution could reflect the fundamental role of the Seeker in modelling the conversation. In the first phase of exploration the Seekers' personal opinions are explicitly elicited by the Recommenders' inquiries. Instead, in the exploitation phase, the Seeker could also provide *soft* evidence of their preferences, which may be used by the Recommender to help the Seeker find a suitable item. This attitude is very common in human-human dialogue with respect to the human-machine interaction, since it follows the principles of cooperative dialogue (Grice, 1975). For this reason, Recommender systems that adopt a proactive behaviour and take the initiative to provide a piece of information that is not explicitly requested, should be able to better achieve the user needs and fulfil the goal of the dialogue (Balaraman and Magnini, 2020).

Acknowledgments

The authors would like to thank Franco Cutugno, who, within his interdisciplinary course on Natural Language Processing, provided a fruitful environment for linguists and computer scientist to join their competences and inspired this work. Also, the authors would like to thank Antonio Origlia for the always ready advice, constructive discussion and his insightful comments on this work.

References

Vevake Balaraman and Bernardo Magnini. 2020. Proactive systems and influenceable users: Simulating proactivity in task-oriented dialogues. In *The 24th Workshop on the Semantics and Pragmatics of Dialogue (WatchDial'20)*.

Carla Bazzanella. 2005. Linguistica e pragmatica del linguaggio. un'introduzione.

Wanling Cai and Li Chen. 2019. Towards a taxonomy of user feedback intents for conversational recommendations. In *RecSys (Late-Breaking Results)*, pages 51–55.

Maria Di Maro. 2021. *"Shouldn't I use a polar question?" Proper Question Forms Disentangling Inconsistencies in Dialogue Systems*. Ph.D. thesis, Mind, Gender and Language, University of Naples Federico II.

Zuohui Fu, Yikun Xian, Yongfeng Zhang, and Yi Zhang. 2020. Tutorial on conversational recommendation systems. In *Fourteenth ACM Conference on Recommender Systems*, pages 751–753.

Chongming Gao, Wenqiang Lei, Xiangnan He, Maarten de Rijke, and Tat-Seng Chua. 2021. Advances and challenges in conversational recommender systems: A survey. *arXiv preprint arXiv:2101.09459*.

Herbert P Grice. 1975. Logic and conversation. In *Speech acts*, pages 41–58. Brill.

Shirley Anugrah Hayati, Dongyeop Kang, Qingxiaoyang Zhu, Weiyan Shi, and Zhou Yu. 2020. Inspired: Toward sociable recommendation dialog systems. *arXiv preprint arXiv:2009.14306*.

Andrea Iovine, Fedelucio Narducci, and Marco de Gemmis. 2019. A dataset of real dialogues for conversational recommender systems. In *CLiC-it*.

Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2021. A survey on conversational recommender systems. *ACM Computing Surveys (CSUR)*, 54(5):1–36.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Fedelucio Narducci, Pierpaolo Basile, Andrea Iovine, Marco de Gemmis, Pasquale Lops, and Giovanni Semeraro. 2018. A domain-independent framework for building conversational recommender systems. In *KaRS@ RecSys*, pages 29–34.

Thuy Ngoc Nguyen and Francesco Ricci. 2018. A chat-based group recommender system for tourism. *Information Technology & Tourism*, 18(1):5–28.

Henry Prakken. 2018. *Historical overview of formal argumentation*, volume 1. College Publications.

Weiyan Shi, Xuewei Wang, Yoo Jung Oh, Jingwen Zhang, Saurav Sahay, and Zhou Yu. 2020. Effects of persuasive dialogues: testing bot identities and inquiry strategies. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–13.

Douglas N Walton and Erik CW Krabbe. 1995. *Commitment in dialogue: Basic concepts of interpersonal reasoning*. SUNY press.

Douglas N Walton. 1984. Logical dialogue-games and fallacies.

Language Transfer for Identifying Diagnostic Paragraphs in Clinical Notes

Luca Di Liello, Olga Uryupina and Alessandro Moschitti
University of Trento, Italy
{luca.diliello,moschitti}@unitn.it, uryupina@gmail.com

Abstract

English. This paper aims at uncovering the structure of clinical documents, in particular, identifying paragraphs describing "diagnosis" or "procedures". We present transformer-based architectures for approaching this task in a monolingual setting (English), exploring a weak supervision scheme. We further extend our contribution to a cross-lingual scenario, mitigating the need for expensive manual data annotation and taxonomy engineering for Italian.

Italian. *In questo lavoro abbiamo studiato approfonditamente la struttura dei documenti clinici ed, in particolare, abbiamo creato sistemi automatici per l'estrazione di paragrafi contenenti diagnosi e procedure. Attraverso l'utilizzo di modelli basati sull'architettura transformer, abbiamo estratto diagnosi e procedure nel setting monolingua (in inglese). Successivamente, abbiamo esteso la nostra ricerca allo scenario multilingue, riducendo il fabbisogno di larghi dataset in italiano annotati manualmente grazie all'utilizzo di machine translation e transfer learning.*

1 Introduction

Big Data approaches have been shown to yield a breakthrough to a variety of healthcare-related tasks, ranging from eHealth governance and policy making to precision medicine and smart solutions/suites for hospitals or individual doctors. They rely on large-scale and reliable automatic processing of vast amounts of heterogeneous data, i.e., images, lab reports and, most importantly, textual medical documentation.

The current paper focuses on *Medical Discourse Analysis*: imposing structure on digitalized health reports through document segmentation and labeling of relevant segments (e.g., `diagnoses`). Identifying and interpreting discourse fragments is essential for accurate and robust Information Extraction from medical documents. In terms of doctor assistance, such a system could quickly and reliably identify the most crucial parts of voluminous health records, allowing to highlight them for improved visibility and thus reducing cognitive load on doctors. For example, a highlighted problematic diagnosis can alert a doctor perusing a large medical dossier. In terms of automated data analytics, discourse structure is crucial for correct interpretation of extracted information. For example, if we want to study a possible correlation between the use of a specific medicine and some outcome, we should only consider documents where this medicine is mentioned as a part of `therapy`, but not as a part of `allergies`.

Some medical documents are generated using task-specific eHealth software imposing certain discourse structure. In Italy, however, there is no single software adopted at either national or regional levels. While there is a general agreement on the nature of information to be included, there are no guidelines or programmatic implementations for structuring it. In addition, historical records, produced before the adoption of recording software, follow the logic of individual doctors and thus show even more variability. We aim therefore at a statistical model that is able to infer the discourse structure without making any assumptions on the recording software.

An important advantage of our approach is its adaptability to new domains (e.g., radiology reports) or languages as well as its robustness in the (highly probable) scenario where new report-generating systems appear at the market.

Several recent studies (Sec. 2) focus on segment labeling for medical records in English. To our knowledge, no approach has been proposed so far to analyze medical discourse structure automatically in other languages, including, most importantly, Italian. The required research is hampered by the lack of resources in other languages, ranging from no data annotated for discourse structure, either for training or for benchmarking, to lack of high-coverage resources, e.g., taxonomies. In our study, we propose a language transfer approach to the problem of medical discourse analysis in Italian. We first investigate possibilities for training robust monolingual models (Sec. 4) and then build upon our monolingual results to transfer the model in another language (Sec. 5).

2 State of the Art

In the past decade, a massive effort has been invested into analyzing automatically textual medical data (clinical notes). The notes' internal logic is crucial for interpreting their underlying semantics, thus enabling better understanding and interoperability. This has given rise to empirical studies on the medical document structure: reliable and interpretable annotation guidelines and systems for automatically segmenting clinical notes and annotating segments with labels such as `allergy` or `diagnosis`.

The most thorough attempt at defining clinical records' structure via a taxonomy of *section headers* has been undertaken by Denny et al. (2008). This study developed SecTag—a hierarchical header terminology, supporting mappings to LOINC and other taxonomies. Table 1 shows some SecTag entries related to `diagnosis` and their parameters relevant for the present study.[1] The SecTag concepts (column 1) are organized hierarchically, with specific diagnoses (e.g., admission or discharge diagnoses) being subnodes (column 2) of the main `diagnosis` concept (SecTag node "5.22"). Different ways of expressing the specific semantics via headers (column 3) are then linked to the corresponding nodes. SecTag advocates a practical data-driven approach, thus listing headers that are not always grammatical (e.g., "admit diagnosis"), provided they are commonly used by practicing clinicians. Most importantly, SecTag goes beyond a superficial view of the task, not only linking easily identifiable headers, (e.g., most common spellings, headers containing important key words), but also organising hierarchically concepts that are normally expressed in very distinct ways (e.g., linking "cause of death" or "gaf" to diagnoses). In total, SecTag provides 94 entries just for `diagnosis`. This shows that a considerable medical expertise is required for creating a similar resource for other languages from scratch.

The SecTag release has led to the development of a related method for automatic identification of sections in clinical notes (Denny et al., 2009), via a combination of NLP techniques, terminology-based rules, and naive Bayes classification.

While the SecTag approach exhibits remarkable performance, creation and maintenance of the header taxonomy is a very expensive task requiring considerable medical expertise. More data-driven approaches have been proposed recently for English (Rosenthal et al., 2019; Dai et al., 2015), among others. These systems, however, require manually labeled data.

3 Data for Identifying Diagnoses and Procedures Segments

3.1 English Data: MIMIC-III

Several large collections of medical data, with partial NLP annotations, have been released recently, for example, MIMIC (Johnson et al., 2016) or I2B2[2]. Unfortunately, none of these resources provide annotation for discourse structure. Our study relies on the MIMIC-III dataset, extending it with an extra layer to label diagnosis and procedure fragments. Our choice follows practical motivations: it is the largest available dataset, most commonly used by the AI community. We only rely on the textual data from MIMIC discharge notes (the NOTESEVENTS table), however, a future work can explore possibilities of joint modeling of textual and numeric data (e.g., lab measurements).

We have built a rule-based algorithm for annotating MIMIC with diagnosis/procedure fragments. We segment a note into fragments and label them based on the headers, looking them up in SecTag (Section 2). For fragments with no header, we propagate the label from the previous fragment. Fragments with headers not

[1] SecTag entires contain 16 parameters, inheriting information from referenced taxonomies such as LOINC, most of them are of no practical relevance in our case and, moreover, are typically set to NULL.

[2] `https://www.i2b2.org/`

concept	taxonomy tree id	header
diagnoses	5.22	diagnosis
principle_diagnosis	5.22.39	primary diagnoses
diagnosis_at_death	5.22.41	cause of death
admission_diagnosis	5.22.44	admit diagnosis
discharge_diagnosis	5.22.45	discharge_diagnosis
global_assessment_functioning	5.22.49.58.11	gaf

Table 1: Examples of diagnostic headers in the SecTag taxonomy.

	MIMIC discharge	exprivia-10	exprivia-100
total documents	59652	10	100
paragraphs per doc	30.57	7.7	26.77
diagnoses per doc	1.22	0.8	1.28
documents with no diagnosis	8674 (14.5%)	2 (20%)	27 (27%)
procedures per doc	0.71	N/A	N/A
documents with no procedure	20797 (35.86%)	N/A	N/A

Table 2: MIMIC-III discharge (silver annotation with SecTag) vs. Exprivia datasets (gold annotation).

found in SecTag are considered `−diagnosis`, `−procedure`. The headers are then removed from the document, thus forcing the model to learn paragraph classification from the textual content, relying on headers as a silver supervision signal.

While a typical MIMIC note has a single diagnostic paragraph, some contain multiple diagnostic fragments: (i) some notes span multiple related reports, where each report comes with its own diagnosis; (ii) some notes contain semantically different diagnostic sections (e.g., "admitting diagnosis" and "discharge diagnosis"); (iii) some notes cover complex cases and the diagnostic section is expressed in several (consecutive) paragraphs.

Since SecTag predates major MIMIC releases, some popular headers are missing—we have therefore manually extended the taxonomy (6.7k headers) to cover another 75 of the most popular headers. The expansion yielded a considerable increase in procedure paragraphs, augmenting drastically the number of positive examples for training the `procedure` classifier. At the same time, the overall precision improved, eliminating some consistent errors with diagnosis paragraphs. In what follows, we always rely on data preprocessed with expanded SecTag.

3.2 Italian Data: Exprivia Datasets

A large collection of discharge reports in Italian has been provided by Exprivia S.p.a. The documents show some similarity to MIMIC discharge reports: they are typically 0.5-1 page long, they can be split into paragraphs rather reliably, they exhibit a considerable variability in terms of the underlying discourse structure. Each document is associated with a set of ICD-9 codes for discharge diagnoses. Yet, similarly to MIMIC, no inline manual annotation is provided for identifying textual segments referring to diagnoses/procedures.

To provide accurate test data for our multilingual approach, a human expert has conducted a manual annotation of the Italian set. We have labeled a pilot of 10 notes and a random sample of 100 notes. The annotation only covered `diagnosis` as our pilot phase revealed that labeling `procedure` required considerably more elaborate guidelines and medical training.

Table 2 compares document statistics for discharge notes from MIMIC-III and Exprivia datasets. It suggests that the pilot can only be used as a very preliminary sample of the data: the notes are rather small and with few diagnoses. The Italian documents from *exprivia-100* show a striking similarity to MIMIC: there are on average around 25-30 paragraphs per document, 1.2-1.3 of which are diagnostic. The major difference comes from the documents with no diagnosis (27% in Italian, 14.5% in English). We believe that this similarity reflects the fact that, despite differences in national and local healthcare regulations as well as individual practicing/recording approaches, clinical notes reflect a common underlying semantics and thus a language transfer model can be successful for our task, mitigating the need for very time-consuming and costly expert effort on constructing taxonomies similar to SecTag in Italian.

4 Transformer-Based Architectures for Diagnosis and Procedure Extraction

Transformer-based models have recently become the standard in NLP. Models like BERT (Devlin et al., 2019) and ELECTRA (Clark et al., 2020) showed impressive performance when compared to previous state of the art. These models are based on the Transformer block (Vaswani et al., 2017), which exploits the attention mechanism to find relations between all pairs of tokens in the input text and thus creates deep contextualized representations. Transformer layers can be stacked to create more powerful and refined models. For computational efficiency, we focus on architectures with no more than 12 layers.

Tokenization. Raw text cannot be provided directly to a transformer-based model: it is first tokenized using a fixed-size vocabulary, created via a segmentation algorithm, e.g., *WordPiece*. We extended BERT vocabulary to account for eventual deidentified medical input.

Pre-training and fine-tuning. Transformer-based models are usually trained in a 2-step fashion. The model is first *pretrained* on a huge amount of artificially labelled text taken from sources like Wikipedia or CommonCrawl. At the *fine-tuning* stage, the model is adapted to a specific task, e.g., Question Answering or Diagnosis Extraction. Since the model is already able to create good contextualized representations, the fine-tuning requires only a small amount of manually labelled examples. Following the common transformer fine-tuning practices, we classify paragraphs into $\pm$`diagnosis` with a binary classification head on top of the first token output.

5 Language Transfer for Diagnosis Identification

The main bottleneck for NLP on medical data in Italian lies in the lack of annotated data and professionally created resources, similar to SecTag. To mitigate this issue, we advocate a language transfer approach, combining our transformer models (Section 4) with state-of-the-art machine translation (MT).

We investigate three cross-lingual setting. In the baseline set up, we do not perform any translation, relying on BERT's tokenizer and cross-lingual embeddings to learn informative sub-word clues for diagnostic paragraphs.

Transformer	Language	parameters
BERT-base-uncased	English	109M
BERT-base-cased	English	108M
ELECTRA-small	English	13M
BERT-Ita	Italian	110M
BERTino	Italian	68M

Table 3: Transformer models used in empiric evaluation

Our second cross-lingual pipeline builds directly upon the model presented in Section 4. We use an MT component to translate test documents from Italian into English, run our diagnosis identification model and then port the results to the Italian original via a trivial paragraph-level alignment. Note that this model is trained on high-quality data in English and tested on noisy automatically translated data.

For the third pipeline, we first translate the whole training set from English into Italian, while keeping paragraphs aligned. We follow the methodology from Section 4 to train a new model, operating on Italian directly. Note that, unlike the second pipeline, this approach implies training on noisy automatically translated data while testing on high-quality Italian. The effect of this is two-fold: on one hand, the task becomes more difficult to learn, on the other hand, the resulting classifier should be more robust.

To obtain a satisfactory translation using open-source architectures, we rely on the transformer encoder-decoder models (Tiedemann and Thottingal, 2020) trained on the OPUS corpus[3]. While the OPUS corpus is not tailored specifically to the medical domain, its large size and generic nature allow for training very robust MT models. We exploit the two models to translate from English to Italian [4] and from Italian to English [5]. Both are transformer encoder-decoder models trained with the Causal Language Modeling objective.

6 Experiments

6.1 Setup

Data processing. We split the MIMIC III discharge dataset into training, development and test-

[3]https://opus.nlpl.eu
[4]https://huggingface.co/Helsinki-NLP/opus-mt-en-it
[5]https://huggingface.co/Helsinki-NLP/opus-mt-it-en

Task	Filt. Accuracy	Precision@1
Paragraph-level granularity		
Diagnosis	92.4	95.9
Procedure	97.1	98.4

Table 4: Diagnosis and procedure discourse segments identification, monolingual setting (English), document-level view: training, fine-tuning and testing on subsets of MIMIC-III discharge.

ing sets (60%, 20% and 20% respectively). We used the first for training all the models presented in this study, while we use the other two for checkpoint selection, hyper-parameter tuning (batch size and learning rate) and evaluating the monolingual model. We used the *exprivia-10* set for validation and *exprivia-100* set for testing in the cross-lingual (language transfer) experiments.

Transformer Models. We run most experiments in two modes: (i) with powerful transformer components comprising a large number of parameters and providing top performance such as BERT (Devlin et al., 2019) and BERT-ita[6] and (ii) with small and efficient transformer models such as ELECTRA small (Clark et al., 2020) and BERTino (Muffo and Bertino, 2020). The objective of this setup was to measure the performance/efficiency trade-off.

Table 3 presents all the used transformer models with the respective number of parameters.

Evaluation metrics. Diagnosis/Procedure classification task shows a very skewed label distribution. For this reason, we approach it from an information retrieval viewpoint, i.e., we rank paragraphs based on their probability of containing a diagnosis. We use Mean Average Precision and Precision@1 to evaluate the ranking quality. The former takes into account the whole ranking and is therefore the best indicator of the ranking quality. The latter indicates the number of times a correct diagnosis is returned in the first position. To provide a better comparison, we report MAP and P@1 averaging only over the documents that contain at least one diagnosis. We also report model accuracy in recognizing documents with no diagnoses (Filtering Accuracy). This metric was introduced because a relevant fraction of documents did not contain a diagnosis, see Table 2.

[6]https://huggingface.co/dbmdz/bert-base-italian-xxl-cased

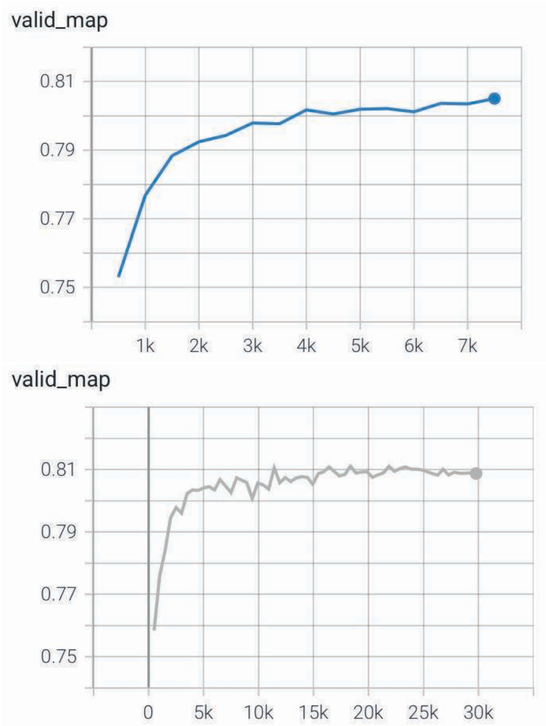


Figure 1: Learning curves on the *exprivia-10* validation set in the Italian pipeline: BERT-Ita (top) vs. BERTino (bottom). MAP (y-axis) for a given number of training steps (x-axis).

6.2 Results

Monolingual results. Table 4 summarizes the English results. The numbers refer to a BERT-base-cased model fine-tuned with a batch size of 64 and a learning rate of $2 * 10^{-6}$. The model is able to identify very accurately documents with no diagnoses/procedures (92.4% and 97.1% accuracy respectively). Moreover, the binary classification of paragraphs into diagnoses (or not), and procedures (or not) is very reliable: 95.9% and 98.4% P@1 at document level.

Cross-lingual experiments. Table 5 shows the results of our language transfer experiments. A moderate performance (58.8% Filtering Accuracy, 49.2% P@1) can be achieved via a BERT model trained on English MIMIC data and directly tested on the Italian *exprivia-100* set. Multilingual-BERT does slightly better as it was trained on 104 languages, English and Italian included. This approach relies on joint multilingual embeddings and fine tokenization. It can, for example, identify and align stems of Latin origin for some disease names. However, it cannot go much beyond: it is not able to model deep semantics related to medi-

Model	Development set	Test set performance		
		Filt. Accuracy	Precision1	MAP
Cross-Lingual BERT				
BERT-base-uncased	exprivia-10	58.8	49.2	58.5
Multilingual-BERT-cased	exprivia-10	51.2	73.5	75.6
MT-based pipeline-2, train on English (MIMIC), test on English translation of exprivia-100				
BERT-cased	exprivia-10	31.8 (7.6)	67.4 (6.8)	69.2 (3.3)
BERT-cased	MIMIC dev	53.1 (9.0)	**73.9** (6.6)	**73.3** (4.9)
ELECTRA-small	exprivia-10	**64.6** (9.5)	60.5 (12.6)	71.2 (9.0)
ELECTRA-small	MIMIC dev	54.2 (8.7)	62.4 (11.2)	73.2 (7.9)
MT-based pipeline-3, train on Italian translation of MIMIC, test on Italian (exprivia-100)				
BERT-ita	exprivia-10	69.8 (6.2)	**78.6** (7.3)	81.5 (3.8)
BERT-ita	MIMIC dev	67.1 (7.8)	73.7 (3.0)	77.2 (3.1)
BERTino	exprivia-10	**72.0** (7.5)	74.9 (2.9)	81.9 (2.6)
BERTino	MIMIC dev	67.7 (4.1)	77.3 (2.5)	**83.3** (1.9)

Table 5: Language transfer models, fine-tuning on the MIMIC training set and evaluation on *exprivia-100* test set; boldface indicates the best results. Standard deviation across 5 runs shown in brackets.

cal processes.

The use of MT shows considerable improvement over the baseline. The results suggest a better performance for the setting where the training set is translated into Italian and the diagnosis extraction model is then learned on (noisy) Italian data. Moreover, this approach is much faster when used as a service, as it directly operates on Italian input.

We performed all the MT-based experiments 5 times using random seeds to enable a better statistical assessment of the results. While in general the standard deviation is rather small considering the very small test set, the setting with a translated test set leads to unstable benchmarking, especially for the smaller ELECTRA transformer.

Finally, smaller transformer models, especially BERTino, exhibit very small performance drops compared to larger transformers. This suggests that they are robust enough to capture paragraph-level diagnosis semantics. Therefore, it is possible to run the extraction service with low computational resources, e.g., using CPUs. Figure 1 shows the stability of the learning with translated training data. Small models are able to match the performance of larger models, being also faster to converge. We believe that smaller models overfit less the MIMIC training data, thus providing a final better performance on the Exprivia data. Note that training was stopped after a fixed amount of time for every experiment. BERTino, being smaller, is able to do more steps in the same amount of time.

7 Conclusion

We present a language transfer approach to unraveling discourse structure of clinical notes, focusing on diagnosis and procedure. We combine transformer-based paragraph modeling with state-of-the-art MT architectures in a novel application, that is essential for eHealth big data analytics. Most importantly, our language transfer approach helps mitigate the need for expensive and time-consuming medical resource creation (annotated train data as well as header taxonomy) in Italian.

We empirically investigate two translation-based architectures, showing that both of them outperform a generic cross-lingual pipeline. The approach based on translating train data is more robust and efficient (at runtime) compared to translating the test data, yielding more stable performance.

In future, we plan to expand our study to other discourse segments, such as allergy or history. However, our first experiments with procedure segments show that, unlike diagnosis, modeling and even annotating other headers require a more tight collaboration with medical experts.

8 Acknowledgements

The research presented in this paper has been supported by the Autonomous Province of Trento (project CareGenius). The computational power has been provided by the High Performance Computing department of the CINECA Consortium (ISCRA project CareGeni).

References

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators.

Hong-Jie Dai, Shabbir Syed-Abdul, Chih-Wei Chen, and Chieh-Chen Wu. 2015. Recognition and evaluation of clinical section headings in clinical documents using token-based formulation with conditional random fields. *BioMed Research International*, 2015.

Joshua Denny, Randolph Miller, Kevin Johnson, and Anderson Spickard. 2008. Development and evaluation of a clinical note section header terminology. In *Proceeding of AMIA Annual Symposium*, pages 156–160.

Joshua Denny, Anderson Spickard, Kevin Johnson, Neeraja Peterson, Josh Peterson, and Randolph Miller. 2009. Evaluation of a method to identify and categorize section headers in clinical documents. *Journal of the American Medical Informatics Association : JAMIA*, 16(6):806–15.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Alistair E.W. Johnson, Tom J. Pollard, Lu Shen, Li wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3.

Matteo Muffo and E. Bertino. 2020. Bertino: An italian distilbert model. In *CLiC-it*.

Sara Rosenthal, Ken Barker, and Zhicheng Liang. 2019. Leveraging medical literature for section prediction in electronic health records. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4864–4873, Hong Kong, China, November. Association for Computational Linguistics.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Trattamento automatico della lingua a supporto dell'editoria: primi esperimenti con il Devoto-Oli Junior

Irene Dini[1], **Felice Dell'Orletta**[1], **Fabio Ferri**[2],
Biancamaria Gismondi[2], **Simonetta Montemagni**[1]
1. Istituto di Linguistica Computazionale "A. Zampolli" – CNR
2. Mondadori Education
{irene.dini,felice.dellorletta,simonetta.montemagni}@ilc.cnr.it
{fabio.ferri,bianca.gismondi}@mondadori.it

Abstract

English. The paper illustrates the results of a first experiment in which Natural Language Processing was used to support the revision of a children's dictionary, in particular for what concerns style and wording of definitions and the enrichment of the list of lemmas. The results achieved are promising and demonstrate the potential of a synergy to be strengthened in the publishing sector.

Italiano. *L'articolo illustra i risultati di un esperimento all'interno del quale tecnologie di TAL sono state utilizzate a supporto della redazione di un dizionario per bambini, in particolare per quanto riguarda la formulazione delle definizioni e l'aggiornamento del lemmario. I risultati raggiunti sono promettenti e mostrano il potenziale di una sinergia da rafforzare nel settore dell'Editoria.*

1 Introduzione

La consapevolezza delle potenzialità di metodi e tecniche di Intelligenza Artificiale (IA) nel settore dell'Editoria sta diffondendosi rapidamente. Il libro bianco su *The Future Impact of Artificial Intelligence on the Publishing Industry* (2019) riporta i risultati di un'indagine internazionale dalla quale emerge che il 25% delle case editrici intervistate ha già investito in applicazioni di tecniche di IA all'interno di diversi settori, che spaziano dal marketing e la distribuzione alla produzione editoriale.

All'interno dello scenario appena delineato, un ruolo centrale è svolto da metodi e tecniche per il Trattamento Automatico della Lingua (TAL), che sono oggi mature per poter contribuire in modo significativo alle diverse fasi del processo editoriale, permettendo - ad esempio - di indicizzare su base semantica il contenuto informativo di un testo, di monitorarne la complessità e l'efficacia comunicativa in relazione alla tipologia dei destinatari, di guidare la sua eventuale riformulazione, di verificare l'eventuale presenza di plagi, oppure di fornire supporto alle fasi di controllo linguistico e tipografico.

In questo contributo, riportiamo i risultati di un primo e promettente esperimento condotto congiuntamente dalla Casa editrice Mondadori Education e dall'Istituto di Linguistica Computazionale del CNR, all'interno del quale tecnologie di TAL sono state utilizzate a supporto della progettazione della nuova edizione di un dizionario per bambini: il *Devoto-Oli Junior* (DJ). In particolare, sono stati affrontati i temi del controllo, della valutazione e della specializzazione del dizionario rispetto alla platea dei destinatari, cercando di conciliare due prospettive apparentemente in contrasto, l'accessibilità dei contenuti da un lato e la loro informatività dall'altro.

2 Il prodotto Dizionario

Una Casa editrice ha con il proprio dizionario un rapporto complesso: opera di notevole impegno redazionale ed economico; pubblicazione di prestigio e, come usa dire, di *brand positioning*; prodotto con diffusione e profitti calanti.

Sperimentare — modi, tempi, target — sbagliando è un lusso che appartiene al passato; da qui l'esigenza di un approccio più certo, più rapido, senza sprechi: dunque, scientifico-tecnologico. E, come sarà descritto meglio sotto, il TAL avvantaggia una Redazione lessicografica nella costruzione del lemmario, anche in relazione al target di mercato.

Quale che sia l'impostazione lessicografica — positivista, storico-linguistica o dal sapore valenziale —, la scelta del lemmario è, prima di ogni cosa, una faccenda di marketing: non vi è dizionario oggi sul mercato che non sbandieri numero di voci o lemmi, di significati, di neologismi.

È facile comprendere che, al momento dell'acquisto, un lemmario sterminato sia garanzia della capacità del dizionario stesso di risolvere i nostri problemi (almeno quelli lessicali e ortografici, s'intende). Eppure la seduzione di un *universo per ordine alfabetico* si scontra con due ineluttabili problemi industriali: il numero di pagine e il costo. Un libro, inteso come oggetto fisico, ha una sua ergonomicità e ci sono limiti fisici oltre i quali le operazioni di rilegatura divengono insensate e la consultazione sgradita. Vi è poi un rapporto matematico diretto — come ricordano incessantemente i Direttori commerciali — tra numero di pagine e costo: nel mondo di Google e Wiktionary, il prezzo è un affare assai delicato, se non per gli acquirenti istituzionali, certo per le famiglie.

Una sfida particolare è poi un dizionario con un target scolastico di riferimento: se infatti un vocabolario dell'uso ha ambizioni totalizzanti, un vocabolario per la scuola è un'operazione ontologicamente editoriale in quanto si fonda sulla capacità di scegliere e ritagliare un mondo linguistico plausibile e utile.

Operazione non così banale qualora si consideri l'ambivalenza della lingua a cui gli studenti sono esposti: da un lato, lessico di base che impiegano con maggior o minor *proficiency*; dall'altro, lessico disciplinare tecnico e tecnico-scientifico di cui sono comprensibilmente ricchi i testi scolastici (*onnisciente*, *antagonista*, *esarcato*, *tettonica*, *fosfolipidico* ecc.). E, come è facile immaginare, questa ambivalenza investirà sia la scelta delle voci sia la costruzione della singola definizione.

3 La nuova edizione di un dizionario

La progettazione della nuova edizione del DJ si è concentrata su due questioni principali:

i. il linguaggio utilizzato nelle definizioni, la sua complessità ed effettiva accessibilità per l'utenza a cui l'opera è destinata, ovvero bambini in età compresa tra gli 8 e i 13 anni;
ii. il lemmario, la sua verifica e il suo aggiornamento a distanza di quasi dieci anni dalla prima edizione, data alle stampe nel 2012.

3.1 La complessità del linguaggio

La scrittura delle definizioni è un punto cruciale e, in genere, molto caratterizzante di questo tipo di opere. Fin dalla prima edizione quindi ci si è molto concentrati su questo aspetto. Definire le parole, sia quelle comuni e "di base" sia quelle meno comuni, più specialistiche o elevate, con altre parole semplici e accessibili a un'utenza con competenze linguistiche in fase evolutiva richiede molte scelte e un piano di scrittura ben definito.

Dal punto di vista lessicale, in prima battuta, è sembrato naturale cercare di definire le parole selezionate utilizzando soltanto le ca. 7.000 voci del *Vocabolario di Base* (VdB) di Tullio De Mauro. Tuttavia, questo metodo ha mostrato presto i suoi limiti, soprattutto quando si è trattato di definire voci o significati tecnico-scientifici. Inoltre, come è emerso nelle interviste effettuate su campioni significativi di insegnanti, i docenti cercano in un dizionario uno strumento didattico che in primo luogo consenta loro di aumentare le competenze lessicali degli alunni, oltre che potenziare quelle già possedute.

Da qui la scelta di utilizzare nelle definizioni qualche parola in più rispetto a quelle del VdB. Coerentemente con questa decisione, ad esempio, nelle definizioni esclusivamente sinonimiche, tipiche degli aggettivi dove l'uso delle perifrasi spesso complica e appesantisce la spiegazione del significato, sono state impiegate triplette di parole, organizzate in un climax che procede dalla parola semanticamente più vicina al lemma a quella più lontana, ma anche da quella più comune a quella più elevata. Purtroppo non sempre, però, i due criteri coincidono, per cui talvolta la parola a più alta complessità lessicale è anche la prima, essendo quella più vicina di significato.

Un altro esempio ci viene fornito dai demotici, una classe chiusa di lemmi per le cui definizioni in genere si approntano delle formule fisse. Proprio a causa della loro ripetitività, queste voci sono sembrate quelle giuste per azzardare l'uso di una parola non comune come *nativo*, inserita nella breve definizione formulare "Abitante, nativo di…", contando anche sulla trasparenza del termine *nativo*, facilmente collegabile a *nato*. Così, in lemmi come *napoletano* troviamo definizioni brevi, come appunto "Abitante, nativo di Napoli", che introducono l'utente a una parola nuova.

C'è poi il problema della complessità sintattica delle definizioni, che merita una riflessione preliminare. Le definizioni dei lemmi di un dizionario obbediscono a regole precise (verbi definiti con verbi, sostantivi con sostantivi, aggettivi con aggettivi o perifrasi attributive, ecc.): Inoltre, per ragioni di spazio, le frasi definitorie sono spesso ellittiche; nel DJ i due casi più frequenti di definizioni ellittiche sono: i) "Abitante di Napoli", dove il determinante è privo di determinato; ii) nei verbi intransitivi, è spesso indispensabile specificare chi è il potenziale soggetto, utilizzando formule tipo "Di mezzo di trasporto, procedere".

Per quanto si sia cercato di evitare le formule ellittiche più pesanti, è chiaro che la complessità sintattica di queste frasi costituisce una delle questioni più spinose da affrontare.

3.2 Il lemmario

I dizionari pensati per questo target sono in genere costituiti da un numero di voci compreso tra un minimo di ca. 15.000, come il *Dizionario Italiano di Base* di Tullio De Mauro (DIB), e un massimo di ca. 23/25.000, come il DJ. Si tratta quindi di repertori lessicografici estremamente selettivi, risultato di scelte molto meditate.

Nel caso del DJ, si è partiti dalle ca. 7.000 voci del VdB, che includono 1.991 parole fondamentali, ca. 2.750 di alto uso e ulteriori 2.337 appartenenti al vocabolario ad alta disponibilità. Grazie a questo primo nucleo, fin dalla prima edizione del DJ sono stati poi lemmatizzati:

i. i derivati più comuni delle 7.000 parole non compresi nel VdB, in modo da fornire agli studenti famiglie di voci il più possibile complete;
ii. molti sinonimi o contrari, utili per collocare ciascun lemma all'interno di una rete cognitiva di collegamenti che ne favorisca la reciproca comprensione e memorizzazione;
iii. i termini non inclusi tra i lemmi del VdB, ma necessari per definirli senza dover ricorrere a complicati giri di parole. Com'è noto, infatti, un dizionario è un sistema chiuso, per cui ogni parola utilizzata per definire deve essere a sua volta definita all'interno dell'opera.

Tuttavia, parole come *sostantivo*, *transitivo* o *coordinata* e *sottrazione*, non incluse nel VdB, rischiavano di non rientrare nel corpus del DJ anche seguendo gli altri criteri individuati. Termini specialistici e disciplinari "di base" come questi non potevano non essere presenti in un dizionario progettato per essere impiegato da insegnanti della scuola primaria e della secondaria di primo grado. L'individuazione dei termini settoriali adatti a questa utenza per numero e livello di specializzazione è dunque il vero nodo da sciogliere. In occasione della prima edizione la soluzione è stata trovata facendo lo spoglio dei manuali delle varie materie della scuola secondaria di primo grado corredati da glossari, un metodo che richiede un considerevole dispendio di risorse e non garantisce risultati soddisfacenti.

4 Il ruolo del TAL nella revisione del DJ

Nella progettazione della nuova versione del DJ, sono state utilizzate tecniche avanzate di TAL a supporto i) del controllo e possibile riformulazione delle definizioni, e ii) della revisione ed eventuale integrazione del lemmario. Le analisi sono state condotte sull'intero corpus dei dati del dizionario in formato XML, per un totale di più di 23.000 lemmi a cui sono associate più di 41.000 definizioni. Come passo preliminare, il corpus delle definizioni è stato linguisticamente annotato con LinguA (Dell'Orletta, 2009; Attardi e Dell'Orletta, 2009; Attardi et al., 2009). I livelli di annotazione alla base delle elaborazioni che seguono sono quello morfo-sintattico e lemmatizzazione, e sintattico a dipendenze.

4.1 Analisi delle definizioni

L'analisi delle definizioni ha riguardato due facce delle complessità linguistica, quella lessicale e quella sintattica. Attraverso questo tipo di analisi è stato possibile identificare quali definizioni contenessero termini e/o strutture sintattiche di difficile comprensione.

La complessità lessicale della definizione è stata calcolata in funzione della complessità lessicale delle parole semanticamente piene che vi ricorrono, sia nella forma in cui effettivamente compaiono, sia in relazione al lemma associato. Numerosi sono i fattori che contribuiscono a rendere un termine complesso, che spaziano dalla frequenza, al grado di ambiguità o di astrattezza, alla lunghezza, per menzionarne solo alcuni (cfr. Shardlow et al. (2021) per una rassegna delle caratteristiche connesse alla complessità lessicale). Seguendo Rayner e Duffy (1986), in questo esperimento ci siamo focalizzati sul fattore frequenza.

La complessità dei termini all'interno delle definizioni è stata calcolata in riferimento a un dizionario di frequenza organizzato in classi costruito a partire dal corpus itWaC (Baroni et al., 2009), ad oggi il corpus più esteso esistente per l'italiano. La classe di frequenza di ciascun termine è stata calcolata in base al corpus utilizzando la seguente funzione:

$$C_{CT} = \lfloor log_2 \frac{freq(MFT)}{freq(CT)} \rfloor$$

dove MFT è il termine più frequente del corpus, CT è il termine considerato e *freq* è una funzione che associa ad un termine la sua frequenza assoluta nel corpus (Richter et al., 2015). Le classi di complessità sono state definite in relazione alle forme e ai lemmi: sono 27 per i lemmi (da 0 a 26) e 26 per le forme (da 0 a 25). Partendo dall'assunto che termini di uso comune vengono considerati semplici mentre termini utilizzati raramente vengono considerati difficili, alla classe 0 appartengono i termini (forme o lemmi) più frequenti e quindi più comprensibili, mentre alle classi 25 e 26 appartengono i termini (rispettivamente forme e lemmi) più rari e più difficili.

Oltre alla complessità lessicale, per ogni definizione è stato calcolato un punteggio di complessità sintattica, utilizzando READ-IT (Dell'Orletta et al., 2011), il primo strumento per la valutazione della leggibilità di testi in italiano basato su TAL. READ-IT si basa su un'analisi sofisticata delle strutture linguistiche sottostanti al testo e articolata su diversi livelli di descrizione linguistica. Per calcolare la complessità sintattica READ-IT si basa su un ampio spettro di tratti linguistici (in particolare morfo-sintattici e sintattici desunti a partire dall'annotazione linguistica condotta preliminarmente). La complessità è espressa con un valore compreso tra 0 (semplice) e 1 (difficile).

4.2 Revisione del lemmario

La revisione del lemmario del DJ è stata condotta attraverso una verifica interna volta a identificare se c'erano termini usati nelle definizioni il cui lemma non era definito nel dizionario, e una verifica rispetto a risorse esterne. Come risorse esterne sono stati usati:

- il lemmario del *Nuovo Vocabolario di Base* di Tullio De Mauro (NVdB), pubblicato nel 2016, oltre trent'anni dopo la prima versione (1980), con l'aggiunta di ca. 1.000 parole;
- il lemmario costruito automaticamente a partire dall'analisi di un corpus di testi per bambini selezionati all'interno della produzione scolastica Mondadori, che comprende l'intero curriculo della Scuola Primaria affiancato dalla cosiddetta parascolastica e da libri di narrativa.

Se l'aggiornamento rispetto al NVdB ha riguardato il lessico comune, l'integrazione rispetto al lemmario estratto dal corpus scolastico Mondadori ha invece comportato un aggiornamento terminologico settoriale, dal momento che il corpus, basato sulla produzione del II ciclo della Scuola Primaria, include libri di lettura e sussidiari antropologici e scientifici.

Nel caso della verifica interna (rispetto al corpus delle definizioni) e quella esterna (rispetto al corpus scolastico Mondadori) sono stati utilizzati lemmari costruiti in modo automatico a partire dall'annotazione morfo-sintattica e dalla lemmatizzazione. Confrontando la lista dei lemmi del dizionario e i lemmari di riferimento (VdB e quelli costruiti automaticamente) è stato possibile identificare i lemmi da valutare per l'eventuale inserimento nel nuovo DJ. Questo tipo di analisi ha portato a identificare più di 160 lemmi del NVdB che non facevano parte del lemmario del DJ, e circa 150 lemmi di parole che ricorrevano nel corpus delle definizioni ma non erano definiti. Più consistente è il numero di lemmi ricavati dall'analisi del corpus scolastico Mondadori, che ovviamente richiede un'analisi attenta mirata a discriminare la terminologia settoriale rilevante per un dizionario per bambini.

5 Elaborazioni: alcuni esempi

5.1 Complessità lessicale

Dopo aver associato le classi di complessità a tutte le parole piene, a ogni definizione sono stati assegnati 4 diversi indicatori di Complessità Lessicale (CL) riguardanti i) la CL dei termini più complessi che vi ricorrono, e ii) la media dei valori di CL di tutte le parole piene all'interno della definizione. In entrambi i casi, il valore di CL è stato calcolato in relazione sia alla forma che al lemma.

La Tabella 1 esemplifica gli indicatori di CL associati ad alcune definizioni. I valori associati a Max $CL_{f/l}$ consentono di identificare definizioni in cui compaiono termini particolarmente difficili (CL_f riguarda le forme e CL_l i lemmi) di cui va valutata una possibile sostituzione con termini più semplici. D'altro canto, i valori associati a Media $CL_{f/l}$ forniscono una misura globale della complessità lessicale della singola definizione, calcolata come la media delle classi di complessità di tutte le parole piene della definizione. Le ultime due colonne della tabella esplicitano la forma/lemma corrispondente al valore Max $CL_{f/l}$: è interessante notare come i valori di forma e lemma più difficili possano far riferimento a termini diversi (cfr. definizione del lemma *antipatia*).

Con questo tipo di analisi sono state identificate le definizioni con un alto grado di CL che richiedevano una revisione. Per esempio, nella definizione di *orda* la parola "scalmanate", con CL=19, è stata sostituita con la parola "agitate" (CL=14), rendendo così la definizione maggiormente comprensibile. Nel caso di una definizione sinonimica come quella di *adombrarsi*, è emerso che le classi associate a "offendersi" e "risentirsi" appartengono alla classe di complessità 14, mentre "indispettirsi" alla classe 20. Il sinonimo associato alla classe più alta di CL è stato quindi retrocesso in ultima posizione dopo quelli più usuali, rispettando il climax previsto.

Ci sono poi casi in cui il lessicografo ha ritenuto opportuno non intervenire per diversi ordini di motivi. Ad esempio, perché la definizione conteneva tecnicismi non sostituibili, nonostante ad alto grado di difficoltà di comprensione, come nel caso della definizione di *ovulazione* riportata in tabella.

Termine	**Definizione**	**Max CL_f**	**Media CL_f**	**Max CL_l**	**Media CL_l**	**Forma con max CL_f**	**Lemma con max CL_l**
adombrarsi	Offendersi, indispettirsi, risentirsi.	20	9	16	8,2	indispettirsi	indispettire
antipatia	Sentimento di avversione istintiva.	14	12,3	14	12,3	istintiva	avversione
orda	Insieme di persone rumorose e scalmanate.	19	11,7	17	12,5	scalmanate	scalmanato
ovulazione	Uscita dall'ovario dell'ovulo pronto per la fecondazione.	17	12	18	12,2	ovario	ovario

Tabella 1: Indicatori di complessità lessicale associati a ogni definizione

La Tabella 2 riporta, per ciascuna categoria grammaticale, le medie dei 4 punteggi di CL associati a ogni definizione. Congiunzioni e avverbi risultano essere le categorie grammaticali le cui definizioni sono complessivamente più semplici. Nomi, verbi, aggettivi, pronomi, articoli e interiezioni risultano invece caratterizzati da definizioni maggiormente complesse.

Classe grammaticale	**Max CL_f**	**Media CL_f**	**Max CL_l**	**Media CL_l**
Aggettivo	12	8,4	11,9	8,5
Articolo	12,8	9,8	13	9,4
Avverbio	10,3	8,2	10,5	8,5
Congiunzione	9,9	7,6	10,2	8
Interiezione	12,6	9,6	12,3	9,6
Nome	12,7	9,3	12,6	9,3
Preposizione	11	8,5	11,1	8,8
Pronome	13,1	8,1	13,2	8,3
Verbo	12,4	9,4	11,6	8,9

Tabella 2: CL media per categoria grammaticale

5.2 Complessità sintattica

Grazie ai punteggi di READ-IT assegnati per il livello sintattico, è stato possibile individuare costruzioni ricorrenti di difficile comprensione. In questo studio preliminare, READ-IT è stato usato nella sua versione corrente, addestrata su testi di tipo giornalistico, per cui i punteggi assegnati a definizioni vanno considerati come indicativi, ma non specializzati rispetto alle peculiarità del linguaggio delle definizioni. Nonostante ciò, è stato possibile identificare definizioni contenenti costruzioni complesse da valutare per un'eventuale riformulazione semplificata, ad es. quelle introdotte da sintagmi preposizionali che ne circoscrivono il dominio o il significato. Per esempio, la definizione di *andare* "Di mezzo di trasporto, procedere" ha associato un indice di complessità sintattica (CS) di 0,36 che si è ridotto significativamente trasformandola in "Detto di mezzo di trasporto, procedere" (CS=0,04). Un altro esempio è costituito dalle definizioni dei demotici come *canadese* la cui definizione è passata dalla forma ellittica "Del Canada" alla forma "Relativo al Canada".

6 Conclusioni

In un dizionario della lingua d'uso il parlante deve potersi rispecchiare, perché è contemporaneamente la fonte e il destinatario dell'opera. Se questo è vero per qualsiasi dizionario, a maggior ragione lo è per quelli rivolti al mercato della Scuola Primaria, nei quali tutto deve essere a misura di bambino: le dimensioni del volume e il prezzo, perché la Primaria è la scuola dell'obbligo per eccellenza, il livello di complessità della lingua, che deve essere proporzionato alle conoscenze e ai bisogni dei bambini e dei loro insegnanti. Dal momento che le esigenze sono tanto particolari, in un'opera come il DJ, dunque, è fondamentale l'impiego di tecniche di produzione che siano efficienti. Le tecnologie TAL hanno risposto perfettamente a questa richiesta di efficientamento. La verifica del lemmario esistente mediante lo spoglio di ampi corpora mirati sul target, la classificazione della complessità lessicale e sintattica delle definizioni individuata attraverso l'impiego di uno strumento come READ-IT, l'individuazione delle nuove voci da inserire grazie all'uso incrociato di tutte queste tecniche hanno prodotto in tempi brevi risultati certi e attendibili. Soprattutto hanno consentito al lessicografo di lavorare su obiettivi circoscritti e gerarchizzati, conciliando la prospettiva dell'accessibilità con quella dell'informatività dell'opera. Il lavoro fianco a fianco di redattori e ricercatori, inoltre, ha aperto nuovi ambiti di sperimentazione e di riflessione, come la ricerca di nuovi modelli definitori, più accessibili rispetto a quelli tradizionali.

Bruno Migliorini, ormai molti decenni fa, chiudeva la sua nitida prosa sul vocabolario con un'affermazione sfiduciata: «sull'avvenire della lessicografia italiana non è possibile far presagi». Oggi, grazie a esperimenti come questo, siamo in grado di dire qualcosa di più: il TAL non potrà non essere parte di questo avvenire.

Bibliografia

Giuseppe Attardi and Felice Dell'Orletta. *"Reverse Revision and Linear Tree Combination for Dependency Parsing"*. In: NAACL-HLT 2009 – North American Chapter of the Association for Computational Linguistics – In Proceedings of Human Language Technologies. Association for Computational Linguistics. June Boulder, Colorado, pp. 261 – 264 (2009)

Giuseppe Attardi, Felice Dell'Orletta, Maria Simi and Joseph Turian. "*Accurate Dependency Parsing with a Stacked Multilayer Perceptron"*. In: EVALITA 2009 – Evaluation of NLP and Speech Tools for Italian 2009. Proceedings, vol. Poster and Workshop Proceedings of the 11th Conference of the Italian Association for Artificial Intelligence article n. 14. EVALITA 2009. December, Reggio Emilia, Italy 2009)

Marco Baroni, Silvia Bernardini, Adriano Ferraresi and Eros Zanchetta. *"The WaCky Wide Web: A collection of very large linguistically processed web-crawled corpora"*. Journal of Language Resources and Evaluation 43(3), 209–226 (2009)

Felice Dell'Orletta, Simonetta Montemagni and Giulia Venturi. "*READ-IT: assessing readability of Italian texts with a view to text simplification*". In Proceedings of the Second Workshop on Speech and Language Processing for Assistive Technologies (SLPAT '11), 30 July, Edimburgo, UK (2011)

Felice Dell'Orletta. "*Ensemble system for Part-of-Speech tagging*". In: Evaluation of NLP and Speech Tools for Italian, 2009. Proceedings Evalita 2009 Organizers, 2009. December, Reggio Emilia, Italy (2009)

Tullio De Mauro (a cura di). *Grande dizionario italiano dell'uso* (GRADIT). Torino: UTET (1999-2000)

Tullio De Mauro. *Il Nuovo vocabolario di base della lingua italiana*. Internazionale, disponibile all'indirizzo https://dizionario.internazionale.it/ (2016)

Giacomo Devoto, Gian Carlo Oli. *Il Devoto-Oli junior*. *Il mio primo vocabolario di italiano*. Mondadori Education, Le Monnier (2012)

Keith Rayner, Susan Duffy. *"Lexical complexity and fixation times in reading: Effects of word frequency, verb complexity, and lexical ambiguity"*. Memory & Cognition, 14:191–201 (1986)

Stefan Richter, Andrea Cimino, Felice Dell'Orletta and Giulia Venturi. *"Tracking the Evolution of Written Language Competence: an NLP-based Approach"*. In Proceedings of the Second Italian Conference on Computational Linguistics (CLiC-it), 3-4 December, Trento, Italy, pp. 236-240 (2015)

Matthew Shardlow, Richard Evans, Marcos Zampieri. "*Predicting Lexical Complexity in English Texts*". Manuscript, arXiv:2102.08773v1 [cs.CL] (2021)

White Paper on the Future Impact of Artificial Intelligence on the Publishing Industry. Gould Finch and Frankfurt Book Fair, disponibile all'indirizzo https://www.buchmesse.de/files/media/pdf/WhitePaperAIPublishingGouldFinch2019EN.pdf (2019)

A First Step Towards Automatic Consolidation of Legal Acts: Reliable Classification of Textual Modifications

Samuel Fabrizi, Maria Iacono, Andrea Tesei and Lorenzo De Mattei
Aptus.AI / Pisa, Italy
{samuel,maria,andrea,lorenzo}@aptus.ai

Abstract

The automatic consolidation of legal texts with the integration of its successive amendments and corrigenda might have an important practical impact on public institutions, citizens and organizations. This process involves two steps: a) the classification of the textual modifications in amendment acts and b) the integration within a single document of such modifications. In this work we propose a methodology to solve step a) by exploiting Machine Learning and Natural Language Process techniques on the Italian versions of European Regulations: our results suggest that the methodology we propose is a reliable first milestone toward the automatic consolidation of legal texts.

1 Introduction

Consolidation consists of the integration in a legal act of its successive amendments and corrigenda.[1] Consolidated texts are very important for legal practitioners. However, their maintenance is a tedious task. Some regulatory publishers such as Normattiva[2] provide continuously updated consolidated texts, others such as Eur-Lex[3] do times to times, some other do not. The automation of this process could let institutions to save resources and practitioners to access continuously updated consolidated documents. This achievement would let organizations stay compliant with the normative more easily. The consolidation process involves two main steps: a) the identification and classification of the textual modifications in amendment acts; b) the integration within a single document of the textual modifications identified in the previous step. The first step can be expressed as the automatic classification of textual modifications inside a legal document. In this work, we focus on step a).

Several authors tried to solve this task using standard Natural Language Processing (NLP) techniques. Ogawa et al. (2008) showed that amendment clauses described in the Japanese statutes can be formalized in terms of sixteen regular expressions. Lesmo et al. (2009) tried to identify and classify integrations, substitutions and deletions using a three-step approach: 1) prune text fragments that do not convey relevant information, 2) perform the syntactic analysis of the retrieved sentences, 3) semantically annotate the provision using a rule-based approach based on tree. In this last step, they also used a knowledge base that describes the provisions taxonomy (Arnold-Moore, 1997).[4] Brighi et al. (2008) and Spinosa et al. (2009) followed a similar approach. In both cases, semantic analysis is carried out on the syntactically pre-processed text using a rule-based approach. The difference is related to the starting point of the semantic analysis. The former's system relied on a deep semantic analysis of the textual modifications. The latter started from the shallow syntactically parsed text. Garofalakis et al. (2016) presented a semi-automatic system for the consolidation of Greek legislative texts based on regular expressions. Francesconi and Passerini (2007) defined a module that automatically classifies paragraphs into provision types. Each paragraph is represented using Bag of words either with TF-IDF weighting (Salton and Buck-

[1] Eur-Lex, About consolidation, https://bit.ly/2VFyGhv

[2] Normattiva, https://www.normattiva.it/

[3] Eur-Lex, https://eur-lex.europa.eu/

[4] A legislative provision represents the meaning of a law part from a legal point of view. Obligations, definitions and modifications are specific types of provision.

ley, 1988) or binary weight. The authors showed an experimental comparison of the different representation methods using the Naive Bayes and Multiclass Support Vector Machine (MSVM) models. This paper describes our approach in the classification of textual modifications, namely substitution, addition, repeal and abolition. The proposed approach is based on standard statistical NLP techniques (Manning and Schutze, 1999). Our method involves i) the use of XML-based standards for the annotation of legislative documents, ii) the construction of the dataset assigning a label to each word according to the tagging format used, and iii) the implementation of NLP models to identify and classify textual modifications. We carried out a systematic comparison among several feature extraction techniques and models. The main contribution of this paper is the application of machine learning models to classify textual modifications. In contrast to rule-based or regular expression techniques, our models do not need expert knowledge about the application domain's properties. They try to extract formulas used to introduce a textual modification without the need for an explicit definition of all the formulas. Our approach leads to lower maintenance costs and hopefully increased robustness of the system.

2 Data

We extracted the data from Daitomic[5], a product that contains all the regulations from a set of legal sources encoded automatically in Akoma Ntoso standard format (Palmirani and Vitali, 2011). We collected from this product all the Italian versions of the amendment documents originally extracted from Eur-Lex and we randomly sampled 260 legal documents for manual labelling.

Accordingly to the Eur-Lex web service specifications[6], we identified seven different types of textual modifications:

- *replacement* annotates a substitution which may concern a part of a sentence (expression, word, date, amount) or a whole subdivision of the document (article, paragraph, indent). Usually, this type of textual modification includes also the following subcategories:
  - *from* annotates the replaced words ("novellando").
  - *to* annotates the words that replace the previous ones ("novella").
- *replacement_ref* is a type of replacement. We use it to handle textual modifications that include attachments.
- *addition* annotates textual modifications that add or complete a part of a legal document.
- *repeal* indicates the removal or reversal of a law. It is used to invalidate its provisions altogether.
- *abolition* indicates the removal of a law part. It is used to replace the law with an updated, amended or related law. This textual modification could just involve single words or whole subdivision as in the replacements.

Category	**Total**
replacement	308
from	95
to	95
replacement_ref	34
addition	96
repeal	93
abolition	92

Table 1: Total number of textual modifications for each category

Table 2 reports an example for each of the mentioned categories. Table 1 shows the total number of textual modifications per category. The number of *replacements* examples is greater than that the others types of modifications because substitutions can be introduced by different formulas that determine their specific meaning. Indeed, from a preliminary experiment, we understood that there is a relationship of proportionality between the number of formulas used to introduce textual modifications and the number of examples needed to train the models. For this reason, we needed a different number of examples for each category to train our models.

Given the differences among the nature of each modification type, we preferred to split the original problem into five subtasks, namely:

1. *replacement* classification that also contains the *replacement_ref* category;
2. *addition* classification;

[5]Daitomic, `https://www.daitomic.com/`

[6]Eur-Lex, How to use the webservice?, `https://bit.ly/393qt9Z`

3. *repeal* classification;

4. *abolition* classification;

5. *from_to* classification.

The manual annotation consisted in assigning one label at each token of the selected document for each subtask that indicates if it represents or not a textual modification. We defined three different tagging formats: Inside-Outside-Beginning (IOB), Inside-Outside (IO), Limit-Limit(LL). The first two tagging formats are standard.[7] The last one, instead, uses the prefix "L-" to indicate that the token is either the beginning or end of a textual modification. We adopted a specific tagging format for each model based on our preliminary results. The tagging format was one of the most critical choices to improve model performance.
The dataset used for the last subtask is different. Indeed, the *from* and *to* tags are always enclosed within the *replacement* tags. We could not use any of our tagging formats because their syntax does not permit any nesting (Dai, 2018). Therefore, we decided to change the dataset itself to train the models. We considered only the tokens inside the sentences representing a replacement and tagged them using the aforementioned tagging formats. In this way, we avoided the nesting issue.

2.1 Preprocessing

Each model needs a different preprocessing method to process the raw text legal documents, depending on the feature extractor used. There are only a few preprocessing operations common to all models:

1. substitution of the special characters ≪ and ≫ with the quote marks;

2. substitution of words between quote marks with the special token `QUOTES_TEXT`. This step has allowed us to limit the number of tokens in each paragraph. The words between quote marks often represent a whole article (for example to substitute or to add). We decided to substitute these words with a special token because they are redundant for our task. This consideration permits us to improve the performances of all models. In the *from* and *to* subtask, we avoided substituting the text between quotes because it has led to a performance improvement.

[7] Breckbaldwin, Coding Chunkers as Taggers: IO, BIO, BMEWO, and BMEWO+, `https://bit.ly/3DzuqBc`

3 Experiments

For each task, we gathered the documents that contain one or more occurrences of that specific modification. Then, we split the dataset into a training and a test set. More precisely, we used the 80/20 ratio adopting a stratified technique (Trost, 1986). We used the training set to validate the hyperparameters of each model. Once computed the final models, we made use of the test set to measure their generalization ability. It is important to emphasise that we never used the internal test set before the definition of the final models.
The general pipeline is composed of the following steps:

1. The annotated documents are tokenized.

2. Each token is associated with one label for each category following the tagging formats previously defined.

3. From each token, we extract its representation using either hand-crafted features or character level N-grams or word embeddings. Depending on the model used, both tagging format and features extraction change.

4. We execute the model selection phase exploiting K-fold cross-validation. In our experiments, we set the K parameter to 3 so that validation sets size is reasonable. The purpose of this step is to find the best hyperparameters of each model.

5. For each subtask, we chose the model with the best performance in the previous step.

6. After choosing the best configuration of each model, we computed and compared their performances over the test set.

3.1 Feature Extraction

We applied several feature extraction techniques to figure out which one was the most effective. In this section, we will explain these techniques with an in-depth description. Considering the nature of the task, all the features are extracted at the word level. We define different sets of features according to the models' needs. We logically divided our features into hand-crafted features, n-gram features and word embeddings.

replacement	All’articolo 7 della decisione 2005/692/CE, la data del <replacement> ≪ <from> 31 dicembre 2010 </from> ≫ è sostituita da ≪ <to> 30 giugno 2012 </to> ≫ </replacement>.
replacement_ref	L’allegato II al regolamento (CE) n. 998/2003 è sostituito dal testo dell’ < replacement_ref > allegato </replacement_ref> al presente regolamento.
addition	È aggiunto il seguente allegato: <addition> “ALLEGATO III [...]” </addition>
repeal	Il regolamento (CEE) n. 160/88 è abrogato. <repeal></repeal>
abolition	nel titolo i termini <abolition>“raccolti nel 1980” </abolition>sono soppressi

Table 2: Annotations examples

In the following we list the **hand-crafted features** extracted and their meaning:

- is_upper: boolean value indicating whether the token is in uppercase
- is_lower: boolean value indicating whether the token is in lowercase
- is_title: boolean value indicating whether the token is in titlecase
- is_alpha: boolean value indicating whether the token consists of alphabetic characters
- is_digit: boolean value indicating whether the token consists of digits
- is_punct: boolean value indicating whether the token is a punctuation mark
- pos_val_cg: coarse-grained part-of-speech from the Universal POS tag set (Kumawat and Jain, 2015): the text has been POS tagged with SpaCy Italian model[8]
- is_alnum: boolean value indicating whether all characters in the token are alphanumeric (either alphabets or numbers)
- word_lower: token in lowercase
- word[-3:]: last three characters of the token
- word[-2:]: last two characters of the token

Then, we decided to use a more complex representation. We used a **Count Vectorizer** (Sarlis and Maglogiannis, 2020) computed over all the Italian legal documents contained in EUR-Lex at the date we created it. It converts a collection of text documents to a matrix of n-gram counts. From each set of words, it produces a sparse vector representation that captures a large number (376037) of character n-grams features.

Finally, we decided to use a **word embedding** lexicon as it has been shown that provides good performances in other Italian tasks (De Mattei et al., 2018; Cimino et al., 2018). We tested a few different in-domain and general purpose embeddings lexicons trained using both fastText (Bojanowski et al., 2017) and word2vec (Mikolov et al., 2013), we obtained the best results with fastText pretrained Italian model (Grave et al., 2018). The features extracted from each token do not contain enough information to discriminate the true amendment class. For this reason, we decided to introduce the *sliding window* concept (Dietterich, 2002). It represents a set of tokens that precede and/or follow each token, like a “window” with a fixed size that moves forward through the text. For each feature extraction technique, we introduced two parameters, `window_size` and `is_bilateral_window`. The former indicates the dimension of the window. The latter is a boolean value indicating whether the window considers only the preceding tokens (False) or both preceding and following tokens (True). For example, the sentence “È aggiunto il seguente allegato” with a bilateral sliding window of size 1, becomes ⟨(PAD, È, aggiunto), (È, aggiunto, il), (aggiunto, il, seguente), (il, seguente, allegato), (seguente, allegato, PAD)⟩ where PAD indicates the padding value. The introduction of the sliding window has made it possible to improve the evaluation metric of all models.

3.2 Models

We want to find a fully automatic approach based on the extraction of interesting features. For this reason, we developed a systematic comparison

[8] Spacy, Models, `https://spacy.io/models/it`

among three models: **Support Vector Machine (SVM)** with n-gram features, **Conditional Random Field (CRF)** with hand-crafted features and a **Neural Network (NN)** that uses word embeddings. This latter model is a rather general convolutional network architecture. The inputs of our NLP tasks are the words that compose the sliding window represented as a matrix. Each row of the matrix corresponds to the word embedding representation of one token. We decided to use a convolutional layer given its efficiency in terms of both representation and speed; it permits us to capture local and position-invariant features (Yin et al., 2017) useful for our purpose. Then, we added a Batch Normalization layer. It significantly reduces the training time in feedforward neural networks (Ba et al., 2016). During the experiment phase, we observed that layer normalization offers a speedup over the baseline model without normalization and it stabilizes the training of the model. We have also tried to use a Bidirectional Long Short-Term Memory based model with an additional CRF layer (Bi-LSTM-CRF) to solve our task (Huang et al., 2015). Its application leads to poor performance in terms of scores and speed. The results obtained show the need to solve our task using simple models that are able to discover local patterns.

4 Results

The objective of the evaluation was to define a systematic comparison among the models' performance with respect to F1 macro, precision and recall. In the model selection step, we used the F1 macro score as the evaluation metric since the frequency distribution of the labels turned out to be strongly unbalanced in all the subtasks.

After some preliminary experiments, we fixed the sliding window size and the tagging format for each model. We found that both the CRF and NN models are more inclined to use a bigger sliding window size (5) than the SVM models (1) from a performance-based perspective. We think this difference comes from the *Curse of Dimensionality* problem that could be encountered in the SVM models (Bengio et al., 2005). Concerning the tagging format, we adopted the `LL` tagging for all the models. Our experiments show that it increases the f1 score of about 20 percentage points.

Table 3 reports the mean results among the 3-fold obtained by the best configuration of each model. The CRF outperforms other models in almost all the subtasks. We think that it is due to the nature of this model. Indeed, CRFs naturally consider state-to-state dependencies and feature-to-state dependencies (Lafferty et al., 2001). Once

Subtask	SVM	CRF	NN
Replacement	0.868	**0.881**	0.841
Addition	0.825	**0.852**	0.796
Repeal	0.915	**0.938**	0.924
Abolition	0.823	0.878	**0.939**
From_To	0.748	**0.873**	0.800

Table 3: Average results in terms of F1 macro score obtained in the validation phase

completed the model selection phase, we chose the best model and its configuration for each subtask. We considered both the mean and standard deviation of the f1 metric among the folds. Then, we re-trained the best model on the whole training set. Table 4 reports the results and the average score of the precision, recall and F1 metrics over the internal test set. The precision score is higher than recall in all except one subtask which may be good for an application perspective.

	Model	Prec.	Rec.	F1
Replacement	CRF	0.949	0.864	0.902
Addition	CRF	0.790	0.865	0.823
Repeal	CRF	0.937	0.912	0.924
Abolition	NN	0.951	0.912	0.931
From_To	CRF	0.977	0.841	0.899

Table 4: Precision, recall and F1 scores of the best model for each subtask

The models' performances are improved compared to the results achieved in the model selection phase, probably thanks to the larger training set provided.

5 Conclusion

We presented and analysed a machine-learning approach to the problem of the classification of textual modifications. We compared different tagging formats, feature extractor techniques and machine learning models. Our experiments show that the sliding window approach, combined with char count vectorizer or word embeddings, allows the models to capture most of the formulas that introduce textual modifications. Following Occam's

razor principle, we defined simple models that obtained good performances in all the subtasks. Our approach does not need any expertise in the law field since it tries to formalized rules to identify textual modifications. We use different NLP techniques to extract hidden features from the words inside a window.
Results validate our approach in terms of both correctness and stability. They represent the first step to build a fully automatic model capable to identify and integrates textual modifications.

References

Timothy Arnold-Moore. 1997. Automatic generation of amendment legislation. In *ICAIL '97*, pages 56–62, 01.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Yoshua Bengio, Olivier Delalleau, and Nicolas Le Roux. 2005. The curse of dimensionality for local kernel machines. *Techn. Rep*, 1258:12.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Raffaella Brighi, Leonardo Lesmo, Alessandro Mazzei, Monica Palmirani, and Daniele Radicioni. 2008. Towards semantic interpretation of legal modifications through deep syntactic analysis. volume 189, pages 202–206, 01.

Andrea Cimino, Lorenzo De Mattei, and Felice Dell'Orletta. 2018. Multi-task learning in deep neural networks at evalita 2018. *Proceedings of the Wvaluation Campaign of Natural Language Processing and Speech tools for Italian*, pages 86–95.

Xiang Dai. 2018. Recognizing complex entity mentions: A review and future directions. In *Proceedings of ACL 2018, Student Research Workshop*, pages 37–44, Melbourne, Australia, July. Association for Computational Linguistics.

Lorenzo De Mattei, Andrea Cimino, and Felice Dell'Orletta. 2018. Multi-task learning in deep neural network for sentiment polarity and irony classification. In *NL4AI@ AI* IA*, pages 76–82.

Thomas G. Dietterich. 2002. Machine learning for sequential data: A review. In Terry Caelli, Adnan Amin, Robert P. W. Duin, Dick de Ridder, and Mohamed Kamel, editors, *Structural, Syntactic, and Statistical Pattern Recognition*, pages 15–30, Berlin, Heidelberg. Springer Berlin Heidelberg.

Enrico Francesconi and A. Passerini. 2007. Automatic classification of provisions in legislative texts. *Artificial Intelligence and Law*, 15:1–17, 01.

John Garofalakis, Konstantinos Plessas, and Athanasios Plessas. 2016. A semi-automatic system for the consolidation of greek legislative texts. In *Proceedings of the 20th Pan-Hellenic Conference on Informatics*, PCI '16, New York, NY, USA. Association for Computing Machinery.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging.

Deepika Kumawat and Vinesh Jain. 2015. Pos tagging approaches: a comparison. *International Journal of Computer Applications*, 118(6).

J. Lafferty, A. McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*.

Leonardo Lesmo, Alessandro Mazzei, and Daniele Radicioni. 2009. Extracting semantic annotations from legal texts. In *HT '09*, pages 167–172, 01.

Christopher Manning and Hinrich Schutze. 1999. *Foundations of statistical natural language processing*. MIT press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Yasuhiro Ogawa, Shintaro Inagaki, and Katsuhiko Toyama. 2008. Automatic consolidation of japanese statutes based on formalization of amendment sentences. In Ken Satoh, Akihiro Inokuchi, Katashi Nagao, and Takahiro Kawamura, editors, *New Frontiers in Artificial Intelligence*, pages 363–376, Berlin, Heidelberg. Springer Berlin Heidelberg.

Monica Palmirani and Fabio Vitali, 2011. *Akoma-Ntoso for Legal Documents*, pages 75–100. Springer Netherlands, Dordrecht.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523.

S. Sarlis and I. Maglogiannis. 2020. On the reusability of sentiment analysis datasets in applications with dissimilar contexts. In Ilias Maglogiannis, Lazaros Iliadis, and Elias Pimenidis, editors, *Artificial Intelligence Applications and Innovations*, pages 409–418, Cham. Springer International Publishing.

Pierluigi Spinosa, Gerardo Giardiello, Manola Cherubini, Simone Marchi, Giulia Venturi, and Simonetta Montemagni. 2009. Nlp-based metadata extraction for legal text consolidation. In *ICAIL*, pages 40–49, 01.

Jan E Trost. 1986. Statistically nonrepresentative stratified sampling: A sampling technique for qualitative studies. *Qualitative sociology*, 9(1):54–57.

Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. 2017. Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*.

Hate Speech and Topic Shift in the Covid-19 Public Discourse on Social Media in Italy

Komal Florio, Valerio Basile, Viviana Patti
Department of Computer Science, University of Turin, Italy
{komal.florio,valerio.basile,viviana.patti}@unito.it

Abstract

The availability of large annotated corpora from social media and the development of powerful classification approaches have contributed in an unprecedented way to tackle the challenge of monitoring users' opinions and sentiments in online social platforms across time but also arose the challenge of temporal robustness of such detection and monitoring systems. We used as case study a dataset of tweets in Italian related to the COVID-19 induced lockdown in Italy to measure how quickly the most debated topic online shifted in time. We concluded that it is a promising approach but dedicated corpora and fine tuning of algorithms are crucial for more insightful results.

1 Introduction

The task of abusive message detection is a very challenging one and from multiple perspective. From the computational point of view, despite the increasing interest and effort of the community on developing automatic systems abusive language detection and related tasks for different languages (Poletto et al., 2021; Vidgen and Derczynski, 2021), the robustness of detection and monitoring systems emerges as a crucial factor to be addressed, where one of the main limitations observed is to consider the Natural Language Processing (NLP) task of detecting abusive language in isolation, without taking into account the intersection with the contextual or social dimensions, that could contribute to a more holistic comprehension of the abusive phenomena in language. In fact, it is becoming increasingly evident that the goodness of hate speech prediction systems, and of NLP algorithms in general, is rooted in how well they capture and model all the relevant characteristics of language in the context of a specific phenomenon and its evolution over time (Jurafsky and Martin, 2000; Nadkarni et al., 2011; Feldman, 2013; Schmidt and Wiegand, 2017; Fortuna and Nunes, 2018). This brought us to intersect our NLP research with the field of Computational Social Science.

The recent availability of long-term and large-scale digital corpora and the effectiveness of methods for representing words over time can play a crucial role in the recent advances in this field. In particular, social media have recently become one of the predominant sources of linguistic data, being the venue for noticeable phenomena in the domain of NLP tasks. They represent the ideal communication context to address the challenges we have outlined.

This paper aims to characterize how the online conversation on the Italian Twitter around the first Covid-19 lockdown, imposed in Italy in 2020, shifted very quickly from one heated debate to another one, following the quick succession of news reports on both news cases and institutional advice and rules on how to navigate everyday life as the crisis was unfolding in the entirety of the world. At first we tried to identify the most polarizing conversation by analyzing the presence of hate speech using AlBERTo (Polignano et al., 2019) but we found that this BERT (Devlin et al., 2019) based algorithm, trained on Italian Social Media language, seemed to under-perform, in comparison with similar case studies (Capozzi et al., 2019). We hence performed the same task using an abusive language computational lexicon, Hurtlex (Bassignana et al., 2018). We discovered the most recurrent types of abusive language, their distribution over time and correlation with real life events regarding the ongoing pandemic.

To identify the most debated topics we resorted to topic modeling and in particular the Dynamic Topic Modeling allowed us to describe how the most frequent topics evolved over time and shed lights on the interplay with the governmental measures that sparked the most debated conversations.

2 Abusive Speech Prediction

In this work we use as case study a dataset of tweets related to the COVID-19 induced lockdown in Italy, as this was the perfect example of government measures that deeply affected everyday life of citizens and hence had the potential to spark very heated debates online. We rely on a recently developed resource, named 40wita [1] (Basile and Caselli, 2020), created by means of filtering with a set of dedicated keywords the publicly available TWITA dataset (Basile et al., 2018), a long term collection of tweets in Italian. The filtering was run from 1^{st} February 2020 to 30^{th} April 2020 and resulted in the collection of 3309704 tweets.

AlBERTo Our first experiment to detect the most debated conversation consisted in a hate speech prediction with AlBERTo, using the same set of hyper-parameters as in (Florio et al., 2020). The findings show a peak of 6% of daily abusive messages around mid February 2020 and at the end of April 2020, while for the rest of the timestamps the rates were much lower (in some cases almost close to zero) than those found in other Twitter-based datasets (see for example (Capozzi et al., 2020)).

Even allowing for the influence of a different context, this finding induced us to conclude that an unknown but not negligible percentage of hateful messages were left undetected. We believe that increasing the training dataset size and quality could lead to better results. For this experiment the data were annotated using guidelines developed for an hate speech detection task, while a set of new guidelines developed specifically for this context could be a significant improvement in the quality of the labelled data. Another possible adjustment relies on the number of annotators and the exploration of the best metric to compute their disagreement, following the latest published work on annotating subjective tasks (Basile et al., 2021).

Hurtlex In order to get a broader insight of the hateful messages in this dataset that were potentially left out by AlBERTo, we performed the same task by means of Hurtlex (Bassignana et al., 2018)[2], a multilingual computational lexicon that contains 17 different categories of abusive language, each of them consisting of a list of characterising words.

The predominant categories of hate speech are represented by tweets containing derogatory words, abusive terms related to moral and behavioural defects, and words indicating cognitive disabilities and diversity. To gain a deeper insight on how this classification has unfolded we analysed which were the most common words that classified a tweet into a specific category. Quite often the words that determine whether a tweet falls or not into a category, and independently on the category, are very generic (e.g., "problema"="problem", "storia"="history") or can assume very different meaning depending on the context (e.g.: "cane"="dog" can be used as a derogatory term or with a neutral meaning), and this contributes in creating a noisy tweets classification. This insight is meaningful in showing why HurtLex presents some struggles in the accuracy of this task. For this reason, the division into pre-defined categories turned out to be not as informative as we were hoping at the beginning. An improvement on this would encompass a manual revision of the list of words for each category, in order to exclude the most generic ones and retain only those which can potentially improve the accuracy of the result. We also conducted a manual revision of all the tweets belonging to the categories with less than 30 tweets, while for the other categories we choose a random sample of 30 tweets, for consistency with the previous case. One of the most interesting findings was that in the category "rci - locations and demonyms", in contrast to the global dimension of the pandemic, our data counter-intuitively showed that the debate was centered strictly around the measures taken in Italy and the differences between national and local rules.

This lexicon-based approach, even though it did not lead to the desired outcome, was nevertheless important to gain more information on our corpus and experience for future directions. In the next sections we will focus on the most powerful classification tool that we employed on this dataset: two

[1] https://osf.io/n39ks/

[2] http://hatespeech.di.unito.it/resources.html

different algorithms for unsupervised topic modeling.

3 Topic Modeling

We implemented two different classification algorithms. At first we run an exploratory topics analysis with a Latent Dirichlet Allocation (or LDA) and then a Dynamic Topic Modeling (or DTM) to better capture the temporal evolution of topics in the discourse.

Latent Dirichlet Allocation The first topic model algorithm that we applied to our dataset is the Latent Dirichlet Allocation, which was first introduced by Blei (Blei et al., 2003). The popularity and versatility of such algorithm relies on the human-interpretable form of the extracted topics and on being, by construction, very robust when deployed on unseen documents.

This model was able to correctly and precisely identify the conversations around the first relevant news around the incoming pandemic. Examples of this include the first restrictions on movements following the first Covid-19 outbreak in Lombardy and Veneto, the national lockdown issued in March and the consequent gradual shift of the conversation towards the difficulties of normal life in such a new context.

As powerful as this model is, it showed a fundamental limit for our perspective and purpose. The relevant topics were punctual but, as expected, not consistent over time because the model was completely re-trained on data from every single week, hence the results for each single time slice were agnostic of the result for every other time slices, and therefore not time-consistent, or comparable, by design. To overcome this issue we implemented a Dynamic Topic Modeling.

Dynamic Topic Modeling The Dynamic Topic Modeling (Blei and Lafferty, 2006) allows to split the datasets into custom time slices and extracts the same exact topics over all of them, thus enabling an analysis on how topics evolve over time.

At first we fine tuned the model by optimizing the perplexity and the coherence score. The first score captures the behaviour of the model towards data which were previously unknown by means of a normalised log-likelihood of a held-out test set. However there are relevant studies (for example (Chang et al., 2009)) proving that perplexity and human judgement not only often do not correlate, but sometimes they even anti-correlate. For this reason a second metric was elaborated: the coherence score, to better model human judgement. This measure captures the degree of semantic similarity between the words related to each single topic (i.e., a measure of the likeness of their meaning). We did not have an annotated corpus that can serve as a training set, hence we only explored the trend of the coherence score with reference to changes in the number of topics, chunksize of data, number of passes and evaluation score. We then concluded for 5 topics and 20 words per topics, as listed in the following Table 1. We chose to leave one topic undetermined ("Topic 1 - Other") to label all the messages that the algorithm struggled to correctly assign to a specific topic.

Topic No.	**Italian**	**English**
Topic 0	quarantena	quarantine
Topic 1	altro	other
Topic 2	lavoro	work
Topic 3	governo	government
Topic 4	sanità	healthcare

Table 1: Topics Extracted using the Dynamic Topic Modeling.

The DTM outputs each unlabelled topic as a list of words with a relevance value. This value, between 0 and 1, represents the probability of a single word to be affiliated with a specific topic. The rationale behind the decision of choosing only 5 topics is that a higher number did not improve the understanding of the corpus as it led to a noisier classification. Each additional topic consisted of a list of words that were either very general in their meaning, or not very close semantically, or both, which made it very difficult to find a topic label that properly represented all the listed tokens.

The most powerful feature of the DTM is that, for each topic, it is possible to rank the most relevant words based on their attached probability value (of referring to the specific topic) and see how they evolve over time. In the following Figure 1, the change in ranking for all the 20 words involved is presented as a coloured heatmap, where the blue values represents words with higher ranking while the red ones are at the lower end of ranking.

There are two main insights we can gain from this visualization. The first one is that topics are lists of pretty common words, which proves how hard of a task topic detection is, because of the complexity and versatility of human language, where general words can be used in different contexts with different meanings. The second insight is that the biggest changes in the word ranking happen within the first time slices. A possible explanation may be traced back to how this dataset was created. The list of hashtags and trends used to filter the tweets was compiled in February and was fixed in time. This means that potentially interesting tweets were left out because they contained hashtags that emerged as relevant later in time but hence were not captured by the keywords used for selecting relevant tweets.

In order to measure the temporal trend of predominance for each topics, we computed, for each of the 13 time slices, the ratio of documents labeled as predominantly referring to each of the topic.

We plotted in Figure 2 the normalized share of documents classified as containing each of the topics in each time slices, to highlight the relative trends over time.

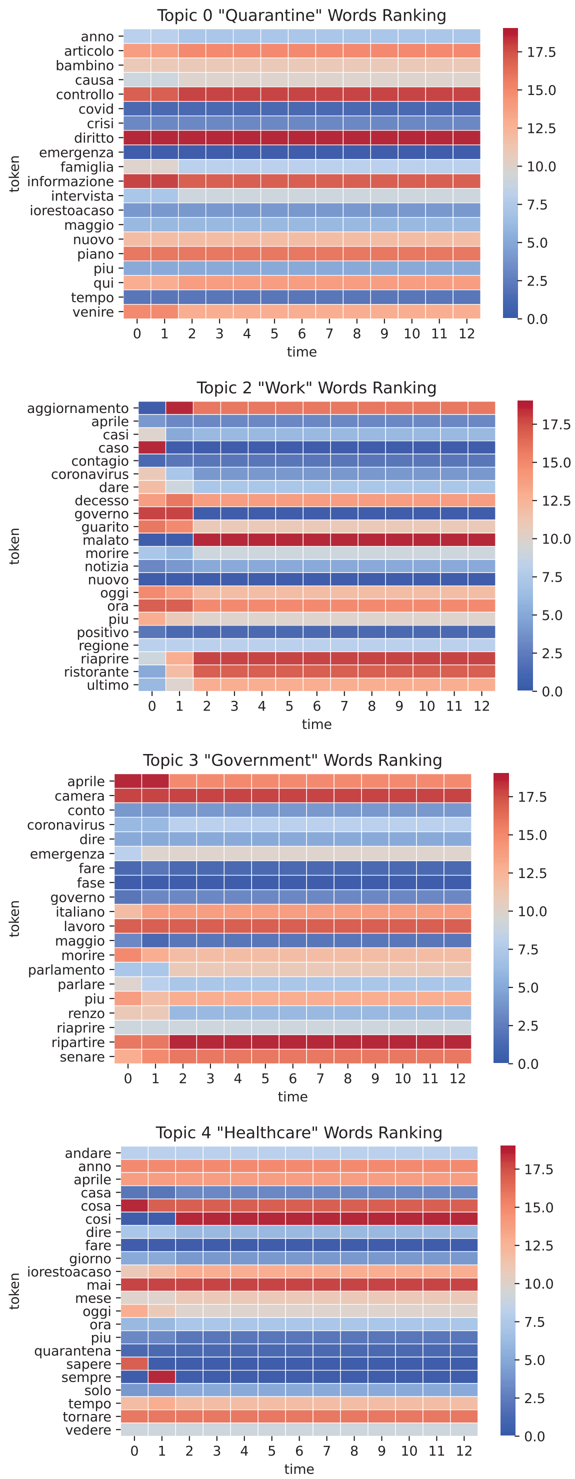


Figure 1: Time evolution of words relevance ranking for all the topics.

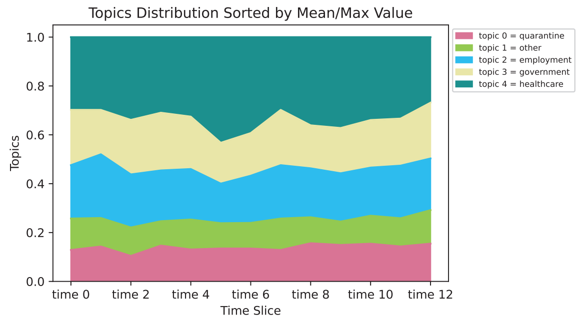


Figure 2: Evolution over time of mean and maximum values of the share of documents related to each of the topics.
We listed both metrics in the same chart as the difference in values was below our error threshold.

It is aligned with our intuition that the largest share of documents across time refer to topic "healthcare". But more in detail it is interesting to analyse the relation between the timestamp of the spikes and relevant Covid-19 events in Italy, as presented in Table 2.

The spikes in shares of documents related to the most predominant topic "quarantine" do follow temporally major events about public health announcement and measures, as shown in Table 2. This proves the point of this research, which

Topic	Time- Slice	Start Day	End Day	Relevant event
4-healthcare	2	16/2/20	22/2/20	public discussion around the first red zones in Veneto
	5	8/3/20	14/3/20	Announcement of the arrival of a medical task force from Cuba in Lombardy (14/3/20). Appointment of a special consultant for the emergency in Lombardy (16/3/20).
2-work	1	9/2/20	15/2/20	public discourse around the Chinese community in Italy
	5	8/3/20	14/3/20	Announcement of the arrival of a medical task force from Cuba in Lombardy (14/3/20). Appointment of a special consultant for the emergency in Lombardy (16/3/20).
	11	19/4/20	25/4/20	First positive news about the Oxford vaccine AstraZeneca
3-government	2	16/2/20	22/2/20	public discussion around the first red zones in Veneto
0-quarantine	3	23/2/20	29/2/20	first red zones issued in Lombardy and Veneto
	9	5/4/20	11/4/20	Economical measure announced. Public discourse around lifting the strict lockdown measures.

Table 2: Relevant Covid-19 events occurred around spikes in the chart.

is that the discourse on Twitter does not only follow closely the most recent and relevant news but it quickly shifts from one topic to the other. In fact, all major peaks in Fig. 2 are followed by a sharply decreasing trend, indicating an immediate loss of predominance and hence an alternation of the dominant arguments of debates.

We explored in a similar way also the temporal evolution of the share of tweets labelled with the Hurtlex categories.

For each of the time slices we computed the relative frequency of tweets labeled with every categories and then created a stacked plot of their maximum values (shown in Figure 3) and the normalized mean values (shown in Figure 4) of their frequencies, to identify both peaks and categories that were consistently predominant through the time.

The relevance of the Hurtlex category related to derogatory words detected over the whole dataset, as described in Section 2, confirms its validity also at a weekly time granularity, as shown by Figure 3. Looking at the chart as a whole it is important to notice that, as we have already highlighted before, the peaks occur in time slices 3 and 5, which respectively correspond the the issue of the first red zones in Italy and two major public health news regarding Lombardy, the hardest hit region of Italy in the first phases of the pandemic (see Table 2 for details).

It is relevant to notice that these peaks occur exactly in the same time slices as the peaks in Figure 2 for the topics "quarantine" and "healthcare", showing that the most heated debates happened around public measures that affected directly and immediately on both the collectivity ("healthcare") and personal life ("quarantine"). Analysing the mean value of the frequencies, in Figure 4, we can see that categories rank differently from Figure 3. More specifically we see that

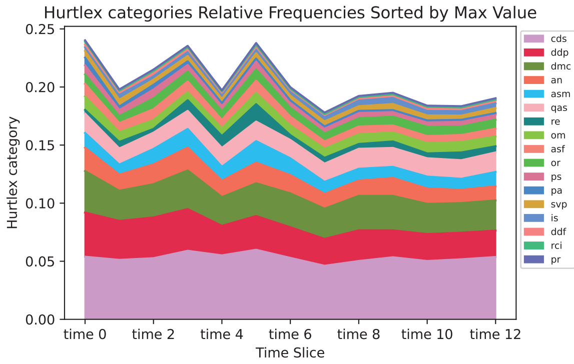


Figure 3: Hurtlex categories maximum frequencies values over time.

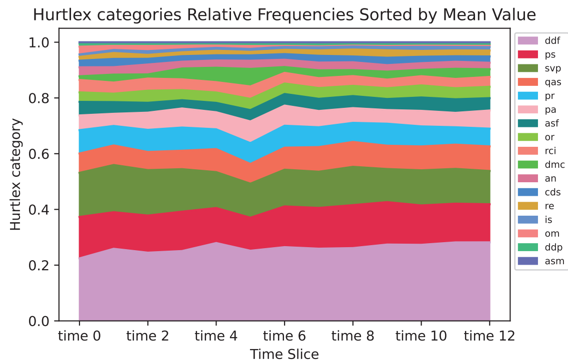


Figure 4: Hurtlex categories mean frequencies values over time.

for example "ddf - physical disabilities and diversity" is by far the most consistent over time but it represents somehow a generic type of offensive language, not correlated with the pandemic, and to some extent this is as a noisy classification of tweets and it would be interesting to investigate further how to improve on this result.

4 Conclusion and Final Remarks

In this work we tried to tackle the challenge of measuring and quantifying the topic shift in the public discourse on Social Media, using as a case study the online debate on Twitter following the Covid-19 related lockdown in Italy in 2020, by means of a dedicated dataset. By combining multiple classification methods we gathered insights into which governmental measures generated the most debated online conversation but we also concluded for the need of deeper investigation on how to build ad hoc corpora and methods to investigate specific linguistic phenomena as online conversation with rapid topic shift following the flow of news coming from both online and traditional media outlets. We also tried to inform AlBERTo with information extracted from topic modeling but the results were far from satisfying. This is a promising way to enhance the accuracy of hate speech prediction, but we concluded that a further investigation on size and characteristics of datasets is essential to gain better results.

References

Valerio Basile and Tommaso Caselli. 2020. 40twita 1.0: A collection of Italian Tweets during the COVID-19 Pandemic.

Valerio Basile, Mirko Lai, and Manuela Sanguinetti. 2018. Long-term Social Media Data Collection at the University of Turin. In *Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018)*, volume 2253 of *CEUR Workshop Proceedings*, pages 1–6, Torino,Italy. CEUR-WS.org.

Valerio Basile, Michael Fell, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, Massimo Poesio, Alexandra Uma, et al. 2021. We need to consider disagreement in evaluation. In *1st Workshop on Benchmarking: Past, Present and Future*, pages 15–21. Association for Computational Linguistics.

Elisa Bassignana, Valerio Basile, and Viviana Patti. 2018. Hurtlex: A multilingual lexicon of words to hurt. In *Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018), Torino, Italy, December 10-12, 2018*, volume 2253 of *CEUR Workshop Proceedings*, pages 1–6. CEUR-WS.org.

David M Blei and John D Lafferty. 2006. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, March.

Arthur TE Capozzi, Mirko Lai, Valerio Basile, Fabio Poletto, Manuela Sanguinetti, Cristina Bosco, Viviana Patti, Giancarlo Ruffo, Cataldo Musto, Marco Polignano, et al. 2019. Computational linguistics against hate: Hate speech detection and visualization on social media in the" contro l'odio" project. In *6th Italian Conference on Computational Linguistics, CLiC-it 2019*, volume 2481, pages 1–6. CEUR-WS.

Arthur TE Capozzi, Mirko Lai, Valerio Basile, Fabio Poletto, Manuela Sanguinetti, Cristina Bosco, Viviana Patti, Giancarlo Ruffo, Cataldo Musto, Marco Polignano, et al. 2020. "contro l'odio": A platform for detecting, monitoring and visualizing hate speech against immigrants in italian social media. *IJCoL. Italian Journal of Computational Linguistics*, 6(6-1):77–97.

Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrish, and David M Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, NIPS'09, page 288–296, Red Hook, NY, USA. Curran Associates Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, MN, USA. Association for Computational Linguistics.

Ronen Feldman. 2013. Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4):82–89.

Komal Florio, Valerio Basile, Marco Polignano, Pierpaolo Basile, and Viviana Patti. 2020. Time of your hate: The challenge of time in hate speech detection on social media. *Applied Sciences*, 10(12):4180.

Paula Fortuna and Sergio Nunes. 2018. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)*, 51(4):85.

Daniel Jurafsky and James H. Martin. 2000. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall PTR, USA, 1st edition.

Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. 2011. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551.

Fabio Poletto, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, and Viviana Patti. 2021. Resources and benchmark corpora for hate speech detection: a systematic review. *Lang. Resour. Evaluation*, 55(2):477–523.

Marco Polignano, Valerio Basile, Pierpaolo Basile, Marco de Gemmis, and Giovanni Semeraro. 2019. AlBERTo: Modeling Italian Social Media Language with BERT. *Italian Journal of Computational Linguistics - IJCOL*, -2, n.2.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10, Valencia, Spain, April. Association for Computational Linguistics.

Bertie Vidgen and Leon Derczynski. 2021. Directions in abusive language training data, a systematic review: Garbage in, garbage out. *PLOS ONE*, 15(12):1–32, 12.

Recognizing Hate with NLP: The Teaching Experience of the #DEACTIVHATE Lab in Italian High Schools

Simona Frenda[1,2], **Alessandra Teresa Cignarella**[1,2], **Marco Antonio Stranisci**[1], **Mirko Lai**[1], **Cristina Bosco**[1] **and Viviana Patti**[1]
1. Università degli Studi di Torino, Italy
2. Universitat Politècnica de València, Spain
{simona.frenda|alessandrateresa.cignarella|marcoantonio.stranisci|mirko.lai}@unito.it
{cristina.bosco|viviana.patti}@unito.it

Abstract

The possibility of raising awareness about misbehaviour online, such as hate speech, especially in young generations could help society to reduce their impact, and thus, their consequences. The Computer Science Department of the University of Turin has designed various technologies that support educational projects and activities in this perspective. We implemented an annotation platform for Italian tweets employed in a laboratory called #DEACTIVHATE, specifically designed for secondary school students. The laboratory aims at countering hateful phenomena online and making students aware of technologies that they use on a daily basis. We describe our teaching experience in high schools and the usefulness of the technologies and activities tested.

1 Introduction

Recently, the presence of digital technologies in our lives has grown enormously, with a strong impact on our daily lives. Digital spaces and social media have become a privileged channel for communication, information and socialization, frequented by millions of people at the same time. Along with the new relational opportunities and access to knowledge, even misbehaviour have acquired new visibility and virality, such as hate speech. In spite of a causal link between hate speech and crime is hard to demonstrate, the risk of offences and effects on psychological and physical well-being of the victims are clear in psychological and social studies (Nadal et al., 2014; Fulper et al., 2014). The extreme consequence of these effects might be suicide, especially considering the adolescents, as suggested by recent studies investigating the link between cyberbullying and suicidal behaviors of U.S. youth (Nikolaou, 2017). To prevent such scenarios, few awareness-raising projects in schools are activated by NGOs in Italy, such as Amnesty International[1] or Cifa ONLUS[2].

The *Commissione Orientamento e Informatica nelle scuole*[3] supports a manifold of activities with the main goal of creating a link between schools and academia, also in the context of the national project *Piano Lauree Scientifiche* (PLS). The members of the CCC (Content-Centered Computing) group of the Computer Science Department of the University of Turin, active in the investigation of hate speech online[4], have led and participated in several hate-speech-related projects, including "Contro l'odio"[5] (Capozzi et al., 2020) a joint work with non-profit entities and University of Bari that aims at monitoring hate speech against minorities in Italy. Within the current experience, we created a data annotation platform specifically dedicated to support educational activities and aimed at reflecting on the importance of a conscientious communication. In this perspective, the idea of #DEACTIVHATE takes hold. This laboratory, addressed at students of secondary schools, is articulated in three main modules with the purpose of:

1) raising awareness about this social problem, encouraging the reflection on microaggressions, hate speech, stereotypes, prejudices;

2) stimulating the so-called *computational thinking* and the study of linguistic elements that are exploited by users to offend or to ex-

[1] http://di.unito.it/silencehateitaly.
[2] http://di.unito.it/iorispetto.
[3] http://di.unito.it/orientamentoscuole.
[4] http://hatespeech.di.unito.it/.
[5] https://controlodio.it/.

press hate against a victim online (hashtags, emoticons, or figures of speech);

3) introducing high schoolers to how tools based on NLP (Natural Language Processing) work to incentivize a more conscious use of technology.

To reach these purposes, We designed a series of educational activities that include: analysis of the online problem by means of an investigation on own social networks personal profiles; linguistic analysis of the hateful messages during the annotation of tweets on the "Contro l'odio" annotation platform; manual identification of hate speech in Italian texts playing the role of 'being an automatic classifier'; translation of this task in a real automatic task, coding two types of classifiers in Python. These activities, delivered online due to the pandemic restrictions, have been distributed in 5 meetings (lasting 2 hours each) for each class, between April and June 2021, for a total of 10 hours per class.

2 Related Work

A popular workshop series on the topic of "Teaching NLP" has been recently held on its fifth edition at NAACL-HLT 2021 (Jurgens et al., 2021), where the participants discussed and shared experiences on a variety of important issues such as: teaching guidelines, teaching strategies, adapting to different student audiences, resources for assignments, and course or program design. The main lesson learned has been that of highlighting the importance of creating materials describing NLP, not only for learners at a university/college level, but also for those learners who are younger and have diverse educational backgrounds. In this regard, a great inspiration for starting to work with schools in Italy derives from the experience of Sprugnoli et al. (2018), where the authors – although with different goal in mind than ours – started a project involving NLP and pupils from Italian schools, aged 12-13. That experience was chiefly dedicated to the study of cyberbullying among pre-teens and the creation of a corpus of WhatsApp threads in the context of the CybeRbullying EffEcts Prevention activities (CREEP) project. Our idea of starting a project that could bring NLP to high schoolers and that, at the same time, could introduce the themes of hate speech, microaggressions, and discrimination by eliciting personal experiences and students' opinions, is somehow in continuity with that experience.

A second work of great relevance for the creation of our experience, has been the reading of Pannitto et al. (2021), in which the authors point out, for the first time, the fact that no high school curricula in Italy includes any *(computational) linguistics* education and that the lack of this kind of exposure makes choosing computational linguistics as a university degree unlikely. Furthermore, the authors highlight that NLP is, indeed, at the core of many tools young people use in their everyday life, and having almost zero knowledge of this field makes the use of such tools less responsible than it could be. The authors have been the first to create a dedicated workshop for Italian, aimed at raising awareness of Italian students aged between 13 and 18 years regarding the subject of NLP (Messina et al., 2021).

Additionally, the idea of creating some playful and meaningful activities regarding NLP and the themes of hate speech for high schoolers, are in line with the concept of '*gamification*', which lately has been applied to many linguistic annotation tasks, as an alternative to crowdsourcing platforms to collect annotated data in an inexpensive way (Bonetti and Tonelli, 2020), such as our "Contro l'odio" annotation platform.

3 #DeactivHate

The goals of #DeactivHate are: 1) raising awareness about misbehaviour online, such as hate speech, eliciting also personal experiences, 2) stimulating computational thinking and linguistic observation of hateful messages, and 3) encouraging a conscious use of technologies discovering how they work. To reach these objectives we articulated three modules as described below.

3.1 Hate Speech: Introduction

The first module aims at introducing a definition of hate speech to students. Hate speech is often mistaken for a generic insult rather than a specific phenomenon "connected with hatred of members of groups or classes of persons identified by certain ascriptive characteristics (e.g., race, ethnicity, nationality)" (Brown, 2015).

The session started with an ice-breaking activity in which students presented themselves through an image found online, depicting an aspect of their identity (see Figure 1). We then asked them to tell whether they were ever attacked or stigmatized for this characteristic.

Figure 1: Example of Jamboard of Google

In this way, we guided the class in drawing a distinction between **non-ascriptive** identity traits (e.g., political belief, style of dressing) and **ascriptive**[6] ones (e.g., ethnicity, sexual orientation, skin colour) (Reskin, 2005). The idea behind this activity is twofold: i) it links issues such as hate speech and racial microaggression (Sue, 2010) to students' lives; ii) it helps distinguishing the spreading of discriminatory contents[7] from generic insults. The module ended with an assignment: students had to find at least one public figure who had been a victim of online discrimination, providing one or more hateful messages as an example, and a counter-narrative response.

3.2 "If I Were a Classifier..."

The second module is organized in two meetings and focuses on the importance of manually annotated corpora for online hate speech detection and what are the peculiarities of hateful messages.

Within the first meeting, each student presented the found messages and try to define the type of attack and the linguistic characteristics of the text that make it hateful or a counter-narrative. The variety of examples led to the introduction of a deeper taxonomy of discrimination (e.g., misogyny, homophobia, sexism, etc...). As expected, the following group discussion brought out a considerable subjectivity in perceiving these phenomena, thus highlighting the need of adopting a shared annotation schema to identify hate speech in messages.

After a brief introduction on what corpora are and how they are used in new technologies, students have been involved in an annotation task of hate speech, asking them to evaluate at least 30 tweets.

For this purpose, we created the data annotation platform[8] within the "Contro l'odio" project. This web application, built using PHP, MySQL, and JavaScript, [9], preserves the student's annotation history by using a passwordless authentication link sent to the email chosen during the login. This method has the twofold advantage of not requiring the student to register to the platform and of preventing ourselves to save the student's email or other personal data. It then ensures the annotation anonymity and satisfies the requirements of General Data Protection Regulation (GDPR), as a desired consequence.

The home page of the web application consists of a dashboard that provides the annotation guideline and shows basic information about the student's activity. Indeed, the student could know the number of sessions they completed (each session consists of annotating 15 tweets) and the level of agreement (expressed in percentage) between their annotation and the annotation performed by the automatic model realized in the "Contro l'odio" project. Gamifying the task through this comparison, we provide the basis for a discussion about the fallibility of automatic systems. Furthermore, we also allow the student to compare their annotation with the annotation of their classmates in order to introduce the measures of annotator agreement. When a session starts, the student could annotate the level of hatefulness of a tweet through a 7 square scale filled with a color scale from *Watusi* to *Sangria* as shown in Figure 2. Two additional squares, respectively filled with *White* and *MidGray*, allow stating the absence of hate or to consider off-topic the content of the tweet. Finally, three toggle switches (on/off button) were added to check the presence of 'irony/sarcasm/humor', 'offensiveness', and 'stereotype', giving them the possibility to reflect about the ways in which users spread hate online.

During the annotation task, students were asked to fill a shared spreadsheet with the tweets that impressed them the most for its offensiveness, for its humorous intention, or the most difficult to anno-

[6] Qualities beyond the control of an individual.

[7] The definition of hate speech we referred to is the one codified by The Council of Europe: "the term 'hate speech' shall be understood as covering all forms of expression which spread, incite, promote or justify racial hatred, xenophobia, anti-Semitism or other forms of hatred based on intolerance" (Recommendation No. R (97) 20).

[8] https://didattica.controlodio.it/.

[9] https://github.com/mirkolai/DEACTIVHATELab.

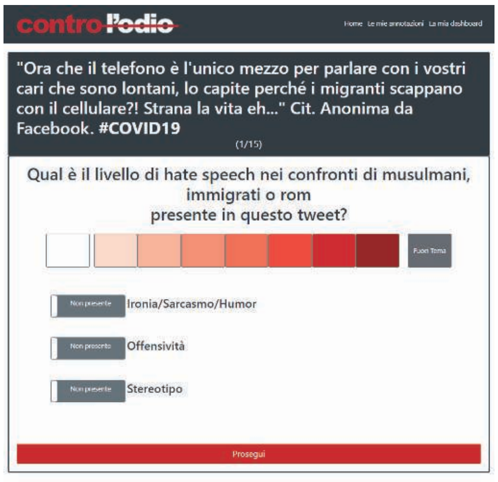


Figure 2: Data Annotation Platform

tate. By discussing with them annotation results, we introduced the latest core concept of the module: the **agreement**. We presented some metrics that are typically adopted to calculate it among annotators and outlined some good practice recently emerged in Corpus Linguistics, such as ensuring the involvement of minorities in corpora development in order to avoid biases (Basile, 2020).

3.3 My First Classifier

In this module the main idea is to stimulate computational thinking by translating linguistic observations coming from the annotation procedure in a proper computational task. The activity of annotation has, indeed, given the opportunity of reflecting on how users tend to verbally express hate online, and on how minorities are represented through stereotypes. To incentivize this transition, we proposed two activities:

A. to mark in each tweet the textual span that could make a classifier aware of the presence of hate speech creating a list of word n-grams;

B. to develop two automatic classifiers (supervised and unsupervised) exploiting the list of word n-grams.

Before starting with the first activity, we asked students to motivate their choice of the tweets selected during the previous exercise. Some tweets triggered a discussions on what should be considered hate speech or not, and the doubts were later solved by looking at the provided definitions of hate speech and at the annotation guidelines. The most controversial tweets report aggressive events or racial propositions; and, for this reason, they were perceived as hurtful by the majority of the students:

(i) *Autobus per i bianchi e altri per i migranti. Non si parla dell'apertheid del Sudafrica né del periodo di segregazione negli Stati Uniti, ma di una proposta della Lega per la provincia di Bergamo. L'Italia non è un paese razzista ma nel 2020 questo è ciò di cui si discute. URL*[10]

Others triggered interesting linguistic reflections, such as:

(ii) *Peccato che non sbarcano povere famiglie africane, ma solo mafia nigeriana, ex galeotti tunisini, stupratori senegalesi, terroristi dell'Isis dalla libia, tutti criminali robusti 1.80 di altezza, pronti a spacciare droga, violentare le nostre donne, cannibali e assassini.*[11]

In these, the students retrieved specific figures of speech such as sarcasm, rhetorical questions and analogies, and also strong words that reflect the social biases towards the minorities. In activity A, all the words and expressions that could make the message hurtful have been collected in a list of n-grams of words called `our_lexicon` (Table 1). Following, the items of such list have been exploited by the classifiers to predict if a tweet contains hate speech or not.

unigrams	risorse, sporchi, pacchia, schifo, invasione, spacciare
n-grams	porti chiusi, cacciarli via, difesa della patria[12]

Table 1: Examples from `our_lexicon`

For activity B, we created an interactive Python notebook using the *Colaboratory* platform provided by Google, as a similar initiative had successfully been carried out by Hiippala (2021) with a similar educational tool. To allow the students to use the notebook in spite of their computer skills, we elaborated some guidelines explaining even how to create a folder in Google Drive and

[10]Translation: *Buses for whites and others for migrants. There is no mention of South Africa's apartheid or the period of segregation in the United States, but of proposal by Lega for the province of Bergamo. Italy is not a racist country but in 2020 this is what we are discussing. URL.*

[11]Translation: *Too bad that poor African families do not land, but only the Nigerian mafia, former Tunisian convicts, Senegalese rapists, ISIS terrorists from Libya, all heavyweight criminals 1.80 tall, ready to sell drugs, rape our women, cannibals and murderers.*

[12]Translation: Unigrams: resources, dirty, *godsend*, disgust, invasion, peddle. N-grams: closed harbours, send [them] away, defence of the fatherland.

how to import all the necessary materials inside of it. Among the required materials, we prepared the dataset using the tweets previously annotated by the students.

We proposed two types of classifiers:

1) unsupervised classifier based on the list `our_lexicon` for which if one of the selected grams are inside the text, the text is predicted as hateful;

2) supervised classifier based on Support Vector Machine algorithm using the list `our_lexicon` as main feature of the classification task.

The coding of the first classifier allowed students to gain confidence with some basics of Python; whereas the second one introduced them to core of new technologies based on machine learning (see Figure 3). At the end of the activity, we observed together the performances of automatic systems and analyzed some of the tweets that were wrongly classified. This final step helped students to reflect on the limitations of machines and the important role of the linguistics in language-related technologies.

4 What We Learnt

Due to pandemic restrictions, we taught the entire laboratory through remote modality (DAD)[13] between April and June 2021 to 2 classes of one secondary school of Turin, with students aged 16-20. As described above, various resources and tools have been used (and created *ex novo*) to bring forward the educational activities in distant teaching mode. However, we plan to propose the same activities/materials even for lessons *in presentia* exploiting the computer rooms of the schools.

For each class, we organized the activities of the three modules in 5 meetings of about 2 hours. Despite the shortness of the laboratory, we found that realizing specific activities for each session helped us manage efficiently the available time. We resorted to web applications to make up for the different devices and operating systems used by the students at their homes. And, in particular, we used Google Meet, as it offers interactive tools such as virtual blackboard, and Moodle, a learning platform provided by the University of Turin that gave us the possibility to organize our activities making available the necessary materials to students. Moreover, each meeting was supported by the use of slides for having visual and descriptive support. The classes assisted in this short period were composed of a total of 35 adolescents, coming from different countries. From the first meeting they showed a general interest in the treated subject, and we were surprised especially by the profoundness of some observations raised during the discussions. The students, indeed, were encouraged to share their opinions, doubts, and perspectives. These discussions made clear that the students face these problems related to technology and communication every day, sometimes suffering even the consequences. Hate speech is, indeed, a very sensitive issue and the perception of what is abusive or not, depends on the cultural background of each student. This fact, on the one side stimulated the debates, however, on the other side, it made it difficult for us to find the *ideal* way to share complex concepts and manage specific situations.

At the end of the laboratory, we provided a survey in order to collect the impressions and the opinions of students. Analyzing these surveys, we noticed that the majority of students considered interesting the content of #DEACTIVHATE, but it appears clear that the format online of the laboratory was perceived from students less interactive and fluent, due especially to technical problems when a part of students were in class and other part at home[14]. From our perspective, we noticed an interesting difference between younger and older students. The older were more active during the activities and discussions than the younger. Moreover, we thought that the number of students affected the flow of the debates, especially in the DAD context. We expect that *in presentia* the proposed activities could have a better impact facilitating the interaction.

5 Conclusion

#DEACTIVHATE represents for Italian high schoolers a first step towards the introduction to subjects such as Linguistics and NLP, that are, for the most part, unknown in Italian high schools, in spite of their relevance in everyday technology. Indeed, this kind of laboratory reveals what are the possible hybrid and multidisciplinary applications

[13] Didattica A Distanza.

[14] For the most part of the school year 2020-2021, Italian schools allowed a capacity of 50% inside classrooms.

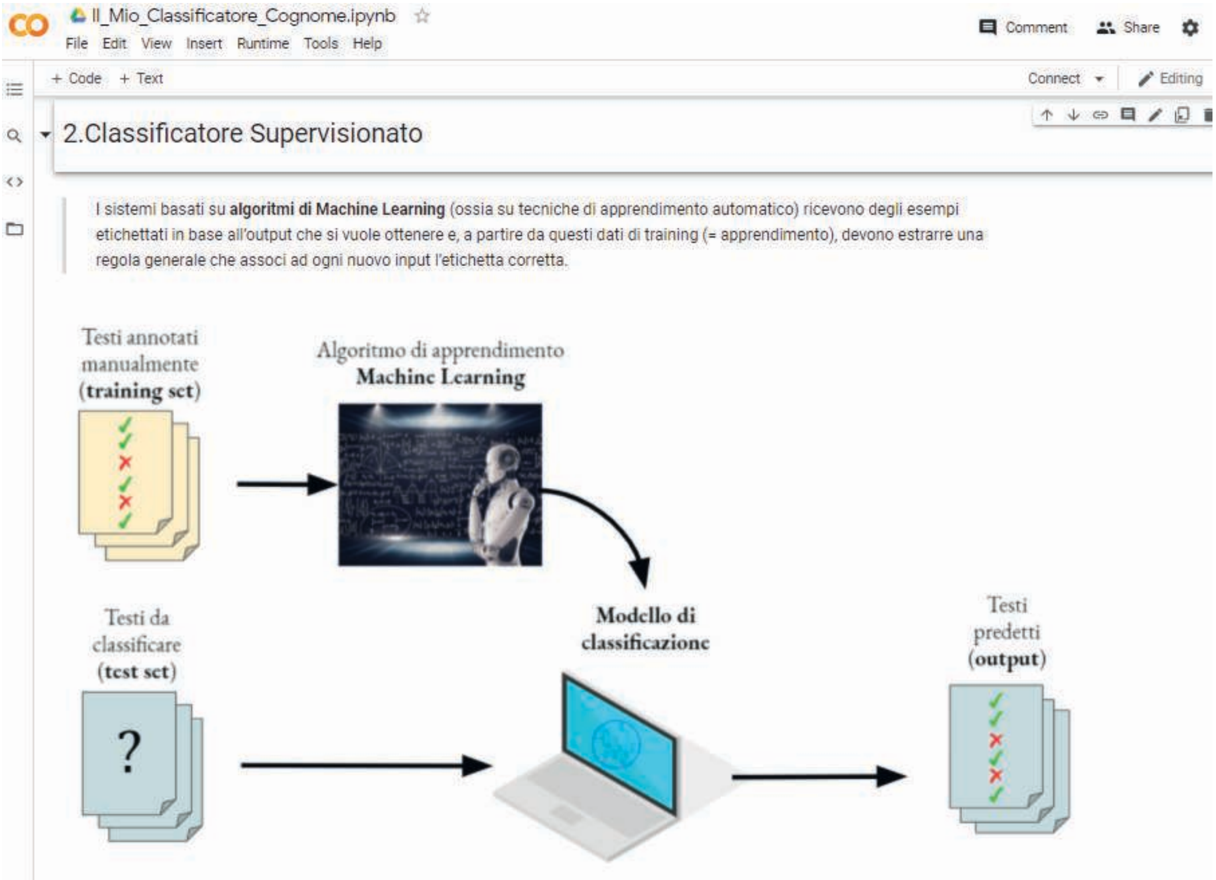


Figure 3: Supervised Classifier Section on Python Notebook

of Computer Science and Linguistics related degrees, far from the *conventional* employment opportunities. Looking at the future, we would like to enhance the proposed activities in order to make them more interactive even in an online context (such as the DAD) following the example of Hiippala (2021).

A final remark needs to be made regarding the lack of evaluative strategies that could allow us to understand the impact of #DEACTIVHATE in students' online behaviors or their knowledge of technologies. Therefore, following the example of Bioglio et al. (2018) and Athanasiades et al. (2015), in the next editions we have planned to employ: surveys before and after the intervention to evaluate the online activity of the students and their experiences about misbehavior (caused or suffered); and interviews to teachers after the conclusion of the laboratory to understand if some changes were perceived with respect to the class group. Future activities will integrate also basic evaluations to assess the degree of learning with respect to the contents of the course, such as computational thinking, annotation methodologies, automatic text processing, as well as a final evaluation of the proposed teaching activities collecting the personal impressions of the students.

In addition, to validate also the impact of #DEACTIVHATE in the society and, in particular, in the city context we think to measure the detection of the amount of hateful message online by means of monitoring platforms, such as the "Contro l'odio" map.[15]

Acknowledgements

The work of S. Frenda, A. T. Cignarella and M. Lai has been funded under the national project *Piano Lauree Scientifiche (PLS) 2019/20* as part of the activities of *Computer Science Department, School of Science of Nature, University of Turin*. The authors would like to extend a special thanks to the school 'Convitto Nazionale Umberto I', and in particular, to Professor Simona Ventura for her availability and her collaboration in this adventure with #DEACTIVHATE.

[15] `https://mappa.controlodio.it/`.

References

Christina Athanasiades, Harris Kamariotis, Anastasia Psalti, Anna C Baldry, and Anna Sorrentino. 2015. Internet use and cyberbullying among adolescent students in Greece: the "Tabby" project. *Hellenic Journal of Psychology*, 12(1):14–39.

Valerio Basile. 2020. It's the End of the Gold Standard as We Know It. In *International Conference of the Italian Association for Artificial Intelligence*, pages 441–453. Springer.

Livio Bioglio, Sara Capecchi, Federico Peiretti, Dennis Sayed, Antonella Torasso, and Ruggero G Pensa. 2018. A social network simulation game to raise awareness of privacy among school children. *IEEE Transactions on Learning Technologies*, 12(4):456–469.

Federico Bonetti and Sara Tonelli. 2020. A 3D Role-Playing Game for Abusive Language Annotation. In *Workshop on Games and Natural Language Processing*, pages 39–43. European Language Resources Association.

Alexander Brown. 2015. *Hate speech law: a philosophical examination*. Routledge.

Arthur TE Capozzi, Mirko Lai, Valerio Basile, Fabio Poletto, Manuela Sanguinetti, Cristina Bosco, Viviana Patti, Giancarlo Ruffo, Cataldo Musto, Marco Polignano, et al. 2020. "Contro L'Odio": A Platform for Detecting, Monitoring and Visualizing Hate Speech against Immigrants in Italian Social Media. *IJCoL. Italian Journal of Computational Linguistics*, 6(6-1):77–97.

Rachael Fulper, Giovanni Luca Ciampaglia, Emilio Ferrara, Y Ahn, Alessandro Flammini, Filippo Menczer, Bryce Lewis, and Kehontas Rowe. 2014. Misogynistic language on Twitter and sexual violence. In *Proceedings of the ACM Web Science Workshop on ChASM*.

Tuomo Hiippala. 2021. Applied Language Technology: NLP for the Humanities. In *Proceedings of the Fifth Workshop on Teaching NLP*, pages 46–48, Online, June. Association for Computational Linguistics.

David Jurgens, Varada Kolhatkar, Lucy Li, Margot Mieskes, and Ted Pedersen, editors. 2021. *Proceedings of the Fifth Workshop on Teaching NLP*. Association for Computational Linguistics.

Lucio Messina, Lucia Busso, Claudia Roberta Combei, Alessio Miaschi, Ludovica Pannitto, Gabriele Sarti, and Malvina Nissim. 2021. A Dissemination Workshop for Introducing Young Italian Students to NLP. In *Proceedings of the Fifth Workshop on Teaching NLP*, pages 52–54, Online, June. Association for Computational Linguistics.

Kevin L Nadal, Katie E Griffin, Yinglee Wong, Sahran Hamit, and Morgan Rasmus. 2014. The impact of racial microaggressions on mental health: Counseling implications for clients of color. *Journal of Counseling & Development*, 92(1):57–66.

Dimitrios Nikolaou. 2017. Does cyberbullying impact youth suicidal behaviors? *Journal of health economics*, 56:30–46.

Ludovica Pannitto, Lucia Busso, Claudia Roberta Combei, Lucio Messina, Alessio Miaschi, Gabriele Sarti, and Malvina Nissim. 2021. Teaching NLP with Bracelets and Restaurant Menus: An Interactive Workshop for Italian Students. In *Proceedings of the Fifth Workshop on Teaching NLP*, Online. Association for Computational Linguistics.

Barbara F. Reskin. 2005. Including mechanisms in our models of ascriptive inequality. *Handbook of employment discrimination research*, pages 75–97.

Rachele Sprugnoli, Stefano Menini, Sara Tonelli, Filippo Oncini, and Enrico Piras. 2018. Creating a WhatsApp Dataset to Study Pre-teen Cyberbullying. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 51–59. Association for Computational Linguistics.

Derald Wing Sue. 2010. *Microaggressions in everyday life: Race, gender, and sexual orientation*. John Wiley & Sons.

The Role of a Computational Lexicon for Query Expansion in Full-Text Search

Emiliano Giovannetti, Davide Albanesi, Andrea Bellandi, Simone Marchi, Mafalda Papini, Flavia Sciolette
Istituto di Linguistica Computazionale, Via G. Moruzzi 1, 56124, Pisa
name.surname@ilc.cnr.it

Abstract

English. This work describes the first experiments conducted with a computational lexicon of Italian in a context of query expansion for full-text search. An application, composed of a graphical user interface and backend services to access the lexicon and the database containing the corpus to be queried, was developed. The text was morphologically analysed to improve the precision of the search process. Some examples of queries are given to show the potential of a text search approach supported by a complex and stratified lexical resource.

Italiano. *Il presente lavoro illustra i primi esperimenti condotti con un lessico computazionale dell'italiano in un contesto di query expansion per la ricerca full-text. È stata sviluppata una applicazione composta da una interfaccia grafica utente e un backend di servizi che permette l'accesso sia al lessico che al database contenente il corpus da interrogare. Il testo è stato analizzato morfologicamente al fine di migliorare la precisione del processo di ricerca. Alcuni esempi di query sono forniti al fine di mostrare le potenzialità di un approccio di ricerca sul testo supportato da una risorsa lessicale complessa e stratificata.*

1 Introduction

The need of techniques going beyond the mere "search by keyword" in the querying of textual resources dates back to the dawn of computational linguistics. Seminal works in the 60s on the development of the very first question answering (QA) systems already included linguistic resources as support datasets. To bring some "old school" examples, the "General Inquirer" QA system (Stone et al., 1962) used a thesaurus for "coding words as to concept membership" while Simmon's "Protosynthex" was equipped with a synonym dictionary (Simmons et al, 1963) to "expand the meaning of the question's words to any desired level". One of the first works specifically focussed on the use of a lexical resource for NLP tasks was about COMPLEX (for "COMPutational LEXicon"), a resource developed at IBM (Klavans, 1988).

The support of linguistic resources has proved its potential in the field of information retrieval (IR) too, as highlighted in many of Bill Woods' works, culminating in the introduction of his conceptual indexing technique and the conceptual taxonomy resource (Woods, 1997) and later refined in an article entitled "Linguistic Knowledge can Improve Information Retrieval" (Woods, et al, 2000). More recently, other researchers have stressed the importance of the availability of a "Lexical Knowledge Base" (another way to refer to a computational lexicon) in tasks such as Word Sense Disambiguation, since their use, in some contexts, can outperform supervised systems (Agirre et al., 2009).

The use of linguistic resources in QA of the earliest period of computational linguistics can be considered as the precursor of "query expansion" (QE), the technique that Manning and Raghavanat describe as the most used "local method" in IR to tackle those situations in which "the same concept may be referred to using different words" (Manning et al., 2008).

Though QE may be obtained in different ways (among which query reformulations based on query log mining) we are here interested in those applications that make use of lexical resources.

Most of the works, published from the 90s to nowadays (proving that QE is still being investigated), exploit WordNet (Fellbaum, 1998), the *de facto* and most widespread ontological (or lexical, depending from the point of view) multilingual resource. Ellen Vorhees was one of the first and used WordNet's IS_A relations to improve text retrieval (Vorhees, 1993). Moving on directly to the most recent works, WordNet has been used with all its ontological features to expand queries in a semantic text search context in (Ngo et al., 2018) while in (Azad and Deepak, 2019) the authors combined WordNet and Wikipedia for QE, exploiting the first to expand individual terms and the second to expand phrase terms.

The research work here illustrated places itself in the context of full-text search carried out using a lexical resource-driven QE technique. However, the focus of this research, differently from that of the cited works, is not on the specific QE technique and the relative evaluation, but on the resource we chose to exploit, introduced in the next section, in place of WordNet and on the frontend and backend technologies implemented to query the text, as described in details in Section 3. The advantages derived from the adoption of a rich and highly structured computational lexicon will also be remarked through some query examples shown in Section 4. The developed application can be freely accessed and used to query the corpus[1].

2 The Context and the Resource

This work stems from the activities conducted by the Institute of Computational Linguistics of CNR (ILC-CNR) in the context of the Talmud Translation Project[2]. The need of providing a way to query the Italian translation of the Talmud[3] on a linguistic basis was the initial spark that led to the idea of experimenting the use of a computational lexicon for Italian. As a matter of fact, this resource (described below) represents a "linguistic mine" which has never been exploited for tasks of full-text search or information retrieval.

2.1 The Parole-Simple-Clips Lexicon

"PAROLE-SIMPLE-CLIPS" (PSC) is a computational lexicon of Italian, developed from 1996 to 2003 by ILC-CNR (Ruimy et al., 2002). Currently, the resource is stored as a MySQL database available on CLARIN[4], and represents a *unicum* among the available linguistic resources for Italian, thanks to its richness and articulated structure of data. Based on the Generative Lexicon theory (Pustejovsky, 1995), the schema on which the linguistic information is encoded is composed of four distinct, but strictly interconnected layers of analysis: phonology, morphology, syntax, and semantics.

In these features lies the motivation of this work, since the available linguistic information may be combined in ways that go well beyond what resources such as WordNet allow to do in the context of text search support. Even considering semantics alone, the information in PSC is detailed with fine-grained features that are not described in WordNet's network of synsets: PSC encodes the meaning of each lexical sense as an array of information, including "templates" (see below), semantic traits, semantic roles, and argumental structures.

In this work, we document the first steps in the use of PSC for QE. At this stage we used: i) the Morphological Units, classified according to their POS, which represent the lemmas of the computational lexicon; ii) the Phonological Units that represent the inflected forms of the lemmas; iii) the Semantic Units (SemUs), that describe the senses expressed by the words. Furthermore, we considered the following morphological and semantic information: i) morphological traits (e.g. gender, number); ii) relations between SemUs (at the moment limited to synonymy and hyponymy); iii) the association between SemUs and "templates", representing sets of senses, labeled according to one of the types represented in the Simple Ontology (Lenci et. al., 2001). The other parts

[1]https://klab.ilc.cnr.it/talmudSearch/

[2]https://www.talmud.it/

[3]The corpus here queried is limited to eight tractates of the babylonian Talmud: Rosh Hashanah, Berakhot, Ta'anit, Kiddushin, Chagigah, Beitza, Sukkah, and Megillah

[4]https://dspace-clarin-it.ilc.cnr.it/repository/xmlui/handle/20.500.11752/ILC-88.

of linguistic information will be the subject of future works, according to an incremental approach.

3 The Process and the Application

The whole search process involves a series of steps that can be summarized as follows (see Fig. 1 for a schematic functional architecture of the application):

i) the user inserts a first set of data to formulate the desired query in the Graphical User Interface;
ii) the interface requests, via Web API, the lexicon backend services which return the linguistic data matching the initial query;
iii) the user completes the query taking into account the linguistic data and starts the search;
iv) the interface executes the query expansion and requests, via Web API, the text backend services which collect, tag, and return the matching textual portions of the Talmud;
v) the interface shows the results to the user.

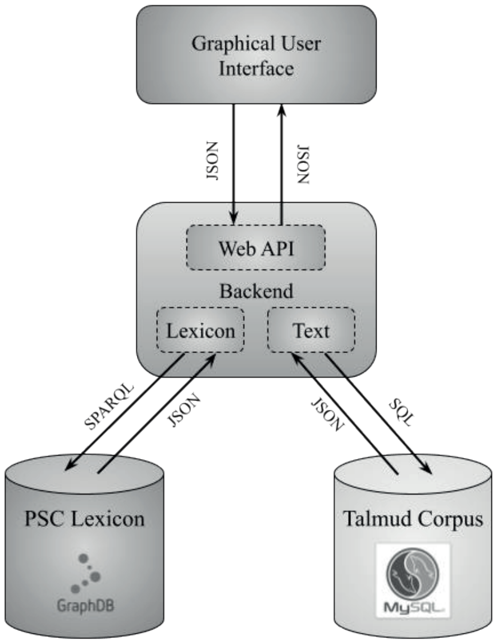


Figure 1. Functional architecture of the application.

First of all, to make the lexicon efficiently queryable, it needed to be transformed from relational data into linked data (Section 3.1). At the same time, a list of services to query both PSC and the database storing the Italian translation of the Talmud needed to be developed in order to answer to the interface requests (Section 3.2). The interface itself was designed on the basis of the available linguistic information exposed from PSC and developed accordingly (Section 3.3). Finally, to improve the precision of the search process, the queried corpus was also POS-tagged (Section 3.4).

3.1 A First Conversion of PSC

The first phase of our work was to consider the relational database of PSC as the data source for the generation of a first Linked Data (LD) conversion. Two main reasons led to the need for a conversion of PSC: i) to ease the reuse of the lexicon itself, in virtue of the intrinsic nature of LD, ii) the possibility of performing automated reasoning on data if appropriately modeled taking into account ontological principles, for example to compute inferred closures, infer new knowledge on the basis of class taxonomies, property hierarchies, and so on. Accordingly to the LD principles, we first had to look for existing vocabularies for the modeling of lexicons.

In the context of the Semantic Web, the *de facto* standard for representing lexical information is the lemon model (Cimiano et al., 2016). Its core module, called OntoLex, allows to represent grammatical, basic morphological and semantic information by means of three main classes: Lexical Entry, Form (lemma and inflected forms), and Lexical Sense. Lemon relies on external vocabularies to define semantic relations between senses: in this conversion we modelled PSC's synonymy and hyponymy with LexInfo ontology[5]. Currently, the converted resource includes 72006 lexical entries (48735 nouns, 6522 verbs, and 11830 adjectives), 469726 inflected forms, and 57130 senses. Explicit lexico-semantic relations include 1803 meronyms, 4060 synonyms, and 44487 hyponyms. This initial conversion of PSC as Linked Data was purely functional to the linguistic querying of the Italian translation of the Babylonian Talmud[6]. Therefore, it was decided to convert a selected number of linguistic data to be exploited for the process of query expansion. At the time of writing this

[5]https://lexinfo.net/
[6]We remark that the conversion of PSC Simple is not the focus of this work, but it was necessary for performing linguistic searches experiments on the Italian translation of the Talmud.

proposal, a complete conversion of PSC as LOD (Linked Open Data) is in progress. This complete conversion will also take full advantage of the already available works on the resource as documented in (Khan et al., 2018) and (Del Gratta et al., 2015).

3.2 Setting up the Backend

Once the computational lexicon was converted, the implementation of the querying system continued with the creation of the backend services needed to access both the lexicon and the database storing the text to be queried. Regarding the lexicon, a GraphDB[7] repository, containing all the converted data, was set up. The access to the repository was implemented with a set of REST services that can be invoked from any web client[8]. The services have been based on the already available backend of LexO, a collaborative web tool for the creation and editing of lemon lexical resources (Bellandi, 2021). At the same time, a list of analogous services was made available to retrieve the textual portions of the corpus matching the expanded queries coming from the frontend of the system. The Italian translation of the babylonian Talmud is currently stored as a MySQL database, where each segment of text appears both in its original and POS-tagged version (see 3.4).

3.3 The Graphical User Interface

The GUI (Fig. 2) set up to query the corpus was developed using Angular[9], one of the most widespread frameworks for frontend Web development, which provides high levels of portability and scalability. In this first version of the search system, the interface was conceived as a sort of "hub" of the whole architecture: from the one side to interact with the user and from the other side to invoke the services exposed by GraphDB and the Talmud database. The interface is divided into two sections. In the left-hand column, the available tractates of the Talmud that can be queried are represented as a tree allowing the user to specify the search context at different levels of granularity. The right-hand section contains the search parameters, where the user can choose between three types of search using the available tabs: Keyword, Form/Lemma, or Semantic Traits.

The first one is the classic keyword-based search. The second type, via the Form/Lemma tab, allows to search for a specific word form or the set of inflected forms of a given lemma by specifying some morphological constraints. By entering a word in the text field, the GUI invokes the lexicon backend services to retrieve the lemmas corresponding to the indicated parameters and displays them with their different senses. Users can then proceed with the search or they can select one or more lemmas and apply to them morphological constraints by clicking on the three dots icon on their right. The selection of at least one of the senses enables the semantic extension search feature: a drop-down menu allows users to look for all the other senses in the lexicon appearing as hypernyms, hyponyms, or synonyms at a specified distance. The forms obtained with this extension are subject to the propagation of the morphological constraints applied to the lexical entry to which they are linked, whether explicit (entered from the interface) or implicit (in the case of a search by form). Finally, the "semantic traits" tab provides two template trees on which multiple selections are possible: the first click selects a template with all its descendants, the second deselects the descendants, and the third deselects the node itself. When the selection changes, the lexicon is queried to obtain the list of senses linked to the chosen templates. Users can then select the desired senses which will be used to retrieve the forms of the relative lemmas to be used in the QE.

All the entered data are used to compose the expanded query, which will be constituted by all the inflected forms provided by the lexicon and matching the indicated morphological constraints, semantic extension, or templates.

The results coming from the backend services accessing the Talmud database are then displayed in a table on the right-hand side, upon which a panel lists the forms retrieved from the lexicon and used for the QE.

[7]Ontotext GraphDB is a highly efficient and robust graph database with RDF/OWL and SPARQL support (https://graphdb.ontotext.com/documentation/free/free/graphdb-free.html)

[8]The source code of the REST services is available at https://github.com/andreabellandi/LexO-backend

[9]https://angular.io/

3.4 POS-Tagging of the Text

For the purpose of reducing the lexical ambiguity in cases where a searched word could match with homographs, the corpus was automatically analyzed and annotated with morphological information.

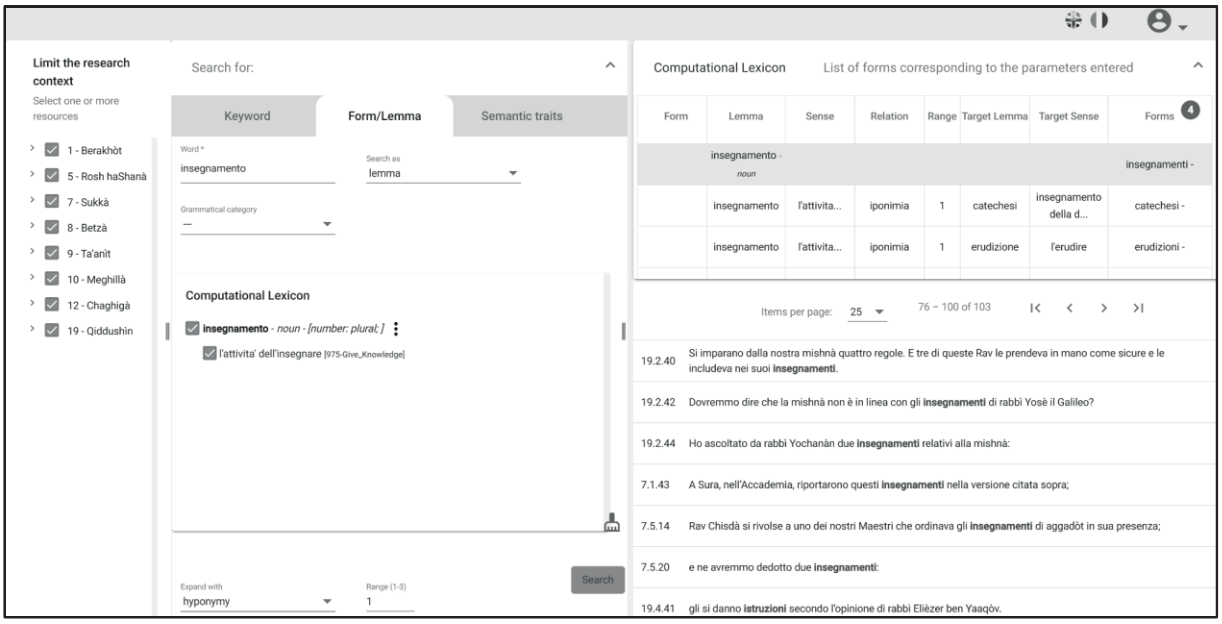


Figure 2. The graphical user interface showing the example of lemma “insegnamento”.

In particular, we parsed all the sentences of the eight tractates of the babylonian Talmud with Stanford's Stanza tools (Qi et al., 2020) using the pre-trained model based on the UD Italian ISDT treebank[10]. The tool was configured to use the processors for tokenization, multi-word token expansion, and Part-of-Speech tagging, which also includes the attribution of morphological traits. Each morphologically annotated textual segment was then stored in the MySQL database to return just the forms matching with the morphological constraints coming from the GUI.

4 Examples of Queries

In this last section, we show a concrete application of the approach by introducing some query examples. Each query can also be tested by the reader by accessing the available application.

The first two examples show the search for words with specific morphological traits and the application of semantic extension. In these cases, the “Form/Lemma” type of search is selected. In the first example, the word “insegnamento” (teaching) is inserted as a lemma. The system finds it in the lexicon and shows it as a noun with one single sense. The user then adds a morphological constraint by setting the “number” trait to “plural”. Finally, the user extends the search to direct hyponyms (distance = 1) and submits the query.

This is a simple case of propagation of the morphological traits through semantics. The lexicon contains the two following key information: i) the fact that the sense of “insegnamento” has three hyponyms: erudizione” (erudition), “istruzione” (instruction), and “catechesi” (catechesis); ii) all the inflected forms and the relative morphological traits of the searched word and its three hyponyms. On the basis of these data, the system composes the final query, which allows to search for all the plural forms of the four lemmas as nouns. As a result, 103 textual segments are retrieved, containing the words “insegnamenti” (97 matches) and “istruzioni” (6 matches) (Fig. 2).

The second example involves the verb “permettere” (to permit/allow), searched as a lemma, with morphological constraints on the finite mood (“indicative”, “subjunctive”, “imperative”, “conditional”). In addition, the user selects just one of the two available senses of the verb (the one with the definition “dare a

[10]https://universaldependencies.org/treebanks/it_isdt/index.html

qlcu la possibilità' di fare qlco" - to give sb the chance to do smth -) and then extends the search to its synonyms. In this case, the lexicon proposes two synonyms of the selected sense: the (single) senses of words "concedere" and "consentire". The resulting expanded query retrieves from the database a total of 405 matches, containing 334 strings of "permettere" (for 131 available forms of the lexicon), 44 strings of "concedere" (for 45 available forms) and 27 strings of "concedere" (for 41 forms).

The last type of search is structured as a more explorative querying of the corpus. In the semantic traits tab, the user can choose one or more between noun/verb or adjectival templates (group of senses), to look for all words relative to a specific semantic field, such as objects, weather verbs, metalanguage, etc.

In this example, the user selects the template "Air animal", which appears as a "leaf" of the sub-tree under the parent-node "Entity". Once the template is chosen, the system retrieves from the lexicon all the relative senses and shows them in a window. It is then possible to select all the available 165 senses or just some of them. Finally, the user can run the search: the system composes the expanded query and retrieves 226 textual segments of the Talmud containing words (both as lemmas and inflected forms) with senses referring to the semantic field of "Air animal": "uccello" (bird), "mosca" (fly), "cavallette" (grasshoppers), and so on.

Among future developments, a feature for a "grouped" selection of multiple templates will be added, that will allow to search for textual segments containing co-occurrences of words referring to the specified templates. To bring an example, the grouped selection of templates "Color" and "Earth animal" will retrieve segments containing multiword expressions such as "vacca rossa" (red cow), "gatta nera" (black she-cat), "oche bianche" (white gooses), etc.

5 Conclusion

As shown in this paper, the availability of a rich and structured linguistic resource (as the computational lexicon we have taken into account) seems to provide an edge over the standard query expansion techniques for full-text search based on WordNet. Now that a very first portion of the resource has been made available (though with a preliminary conversion) and the web application has been implemented, the road is cleared for the next steps.

The first critical issue that will need to be faced involves the limitedness of the resource, covering most - but not all - the lemmas, forms, and senses of standard contemporary Italian and that lacks many domain-related terms or senses. To fill this gap the resource will have to be updated and enriched with more entries.

At the same time, as anticipated, a more in-depth and rigorous conversion of PSC will have to be carried out, a process that will probably take a lot of time and research effort and that for the sake of this first experiment would have been premature and unnecessary. As soon as the whole conversion will be ready, the rest of the information encoded in the lexicon will be made available and integrated in the search process.

Though the benefits of the availability of a computational lexicon wrt WordNet (or a similar resource) may seem obvious in a context of QE for full-text search, an empirical evaluation would be desirable. However, the set up of a benchmark conceived for this purpose appears anything but easy, mainly due to the lack of comparable works or evaluation campaigns focussing on the role of linguistic resources as support.

In conclusion, we believe these first experiments carried out by querying the talmudic text appear promising, especially considering that only a small part of the lexicon has been used. In addition, the support in the disambiguation provided by the POS tagging of the text suggests that an hybridization of a resource-driven QE technique with a deeper stochastic annotation of the corpus to be queried may constitute an interesting experimental field to be investigated.

Acknowledgments

This work was conducted in the context of the TALMUD project and the scientific cooperation between S.c.a r.l. PTTB and ILC-CNR.

References

Eneko Agirre, Oier Lopez de Lacalle, and Aitor Soroa. 2009. Knowledge-Based WSD on Specific Domains: Performing better than Generic Supervised WSD. In *IJCAI'09: Proceedings of the 21st international joint conference on Artificial intelligence*. 1501-1506.

Andrea Bellandi. 2021. LexO: An Open-source System for Managing OntoLex-Lemon Resources. *Language Resources & Evaluation.* https://doi.org/10.1007/s10579-021-09546-4

Ontology-Lexicon Community Group (W3C). Phillip Cimiano, John P. McCrae, and Paul Buitelaar (eds). 2016. *Lexicon Model for Ontologies: Community Report*. https://www.w3.org/2016/05/ontolex/#overview

Riccardo Del Gratta, Francesca Frontini, Fahad Khan, and Monica Monachini. 2015. Converting the PAROLE SIMPLE CLIPS Lexicon into RDF with lemon. *Semantic web* 6: 387-392.

Christiane Fellbaum. 1998. *WordNet: An electronic lexical database.* MA: MIT Press.

Azad Hiteshwar Kumar, and Akshay Deepak. 2019. A new approach for query expansion using Wikipedia and WordNet. *Information sciences* 492: 147-163.

Alessandro Lenci, Nuria Bel, Federica Busa, Nicoletta Calzolari, Elisabetta Gola, Monica Monachini, Antoine Ogonowski, Ivonne Peters, Wim Peters, Nilda Ruimy, Marta Villegas, and Antonio Zampolli. 2000. SIMPLE: A General Framework for the Development of Multilingual Lexicons. *International Journal of Lexicography* 13(4): 249–263.

Fahad Khan, Andrea Bellandi, Francesca Frontini, and Monica Monachini. 2018. One Language to rule them all: Modelling Morphological Patterns in a Large Scale Italian Lexicon with SWRL. In *Proceedings of the 11th International Conference on Language Resources and Evaluation - LREC2018, 2018, Miyazaki, Japan*. hal-01832652

Judith Klavans. 1988. COMPLEX: a computational lexicon for natural language systems. In *COLING '88: Proceedings of the 12th conference on Computational Linguistics*. https://doi.org/10.3115/991719.991802

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*, Cambridge University Press.

Vuong M. Ngo, Tru H. Cao, and Tuan M. V. Le. 2018. WordNet-Based Information Retrieval Using Common Hypernyms and Combined Features. preprint arXiv:1807.05574.

James Pustejovsky. 1995. *The Generative Lexicon.* MA: MIT Press.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In *Association for Computational Linguistics (ACL) System Demonstrations*.

Nilda Ruimy, Monica Monachini, Raffaella Distante, Elisabetta Guazzini, Stefano Molino, Marisa Ulivieri., Nicoletta Calzolari, and Antonio Zampolli. 2002. Clips, a multi-level italian computational lexicon: A glimpse to data. In *Proceedings of the Third International Conference on Language Resources and Evaluation* (LREC02).

Robert F. Simmons, Sheldon Klein, and Keren McConlogue. 1963. Indexing and dependency logic for answering English questions. *American Documentation* 15(3): 196-204.

Philip J. Stone, Robert F. Bales, J. Zvi Namenwirth, and Daniel Ogilvie. 1962. The general inquirer: A computer system for content analysis and retrieval based on the sentence as a unit of information. *Behavioral Science* 7(4): 484–498.

Ellen M. Voorhees. 1993. Using WordNet to disambiguate word senses for text retrieval. In *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval.* https://doi.org/10.1145/160688.160715

William A. Woods. 1997. *Conceptual indexing: A better way to organize knowledge*. Technical Report SMLI TR-97-61, Sun Microsystems Laboratories, Mountain View, CA, April. www.sun.com/research/techrep/1997/abstract61.html.

William A. Woods, Lawrence A. Bookman, Ann Houston, Robert J. Kuhns, Paul Martin, and Stephen Green. 2000. Linguistic knowledge can improve information retrieval. In *ANLC '00: Proceedings of the sixth conference on Applied natural language processing.* https://doi.org/10.3115/974147.974183

A Methodology for Large-Scale, Disambiguated and Unbiased Lexical Knowledge Acquisition Based on Multilingual Word Alignment

Francesca Grasso, Luigi Di Caro
University of Turin, Department of Computer Science
{fr.grasso,luigi.dicaro}@unito.it

Abstract

In order to be concretely effective, many NLP applications require the availability of lexical resources providing varied, broadly shared, and language-unbounded lexical information. However, state-of-the-art knowledge models rarely adopt such a comprehensive and cross-lingual approach to semantics. In this paper, we propose a novel automatable methodology for knowledge modeling based on a multilingual word alignment mechanism that enhances the encoding of unbiased and naturally disambiguated lexical knowledge. Results from a simple implementation of the proposal show relevant outcomes that are not found in other resources.

1 Introduction

Lexical resources constitute a key instrument for many NLP tasks such as Word Sense Disambiguation and Machine Translation. However, their potential may vary widely depending on the nature of the lexical-semantic knowledge they encode, as well as on how the linguistic data are stored and linked within the network (Zock and Biemann, 2020). The resources that are presently available, such as WordNet (Miller, 1995), typically encode lexical-semantic knowledge mainly in terms of word senses, defined by textual (i.e. dictionary) definitions, and lexical entries are linked and put in context through lexical-semantic relations. These relations, being only of a paradigmatic nature, are characterized by a sharing of the same defining properties between the words and a requirement that the words be of the same syntactic class (Morris and Hirst, 2004). Typically related words are therefore not represented due to the absence of syntagmatic links. Additionally, word senses suffer from a lack of explicit common-sense knowledge and context-dependent information. Finally, the well-known fine granularity of word senses in WordNet (Palmer et al., 2007) is due to the lack of a meaning encoding system capable of representing concepts in a flexible way. Other kinds of resources such as FrameNet (Baker et al., 1998) and ConceptNet (Speer et al., 2017) present the same issue, while returning different types and degrees of structural semantic information and disambiguation capabilities.

In this contribution, we provide a novel methodology for the retrieval and representation of unbiased and naturally disambiguated lexical information that relies on a multilingual word alignment mechanism. In particular, we exploit textual resources in different languages[1] in order to acquire and align varied lexical-semantic material of the form $<$*target-concept*, $\{$*related words*$\}^k>$ that are common and shared by all the k languages involved. As we demonstrate through a simple implementation, our method allows to create new lexical-semantic relations between words that are not always available in other resources, as well as to perform an automatic word sense disambiguation process. This system therefore enhances the encoding of prototypical semantic information of concepts that is also likely to be free from strong cultural-linguistic and lexicographic biases.

The benefits provided by our novel multilingual word alignment mechanism are thus fourfold: *(i)* a linguistic and lexicographic de-biasing of lexical knowledge; *(ii)* naturally-disambiguated aligned lexical entries; *(iii)* the discovery of novel lexical-semantic relations; and *(iv)* the representation of prototypical semantic information of concepts in different languages.

[1] In this work, we start with the combination of three languages: English, German and Italian.

2 Background and Related Work

2.1 Bias Types

Due to its complex and fluid nature, lexical semantics needs to undergo a process of abstraction and simplification in order to be encoded into a formal model. As a result, lexical knowledge provided by lexical resources - especially when monolingual - will inherently carry different types of biases. In particular, *i)* linguistic and *ii)* lexicographic biases affect the encoding, consumption, and exploitation of lexical knowledge in downstream tasks.

Linguistic bias Lexical information encoded in a language's lexicon, as well as the potential contexts in which a given lexeme can occur, inevitably reflect the socio-cultural background of the speakers of that language. Lexical resources used for the compilation of lexical knowledge are often conceived as monolingual, therefore they mostly return culture-bounded semantic information which does not account for more shared knowledge.

Lexicographic bias The nuclear components extracted from textual definitions can be different depending on the resource used, even within a single language (Kiefer, 1988). For example, the definition of "*cow*" reported by the Oxford Dictionary is "*a large animal kept on farms to produce milk or beef*" while the Merriam-Webster Dictionary reports "*the mature female of cattle*". Both endogenous and exogenous properties can be subjectively reported (Woods, 1975), such as the term "*large*" and the milk production respectively.

2.2 Related Work

On one side, lexicons are built on top of synsets[2] and contextualize meanings (or senses) mainly in terms of paradigmatic relations. WordNet (Miller, 1995) and BabelNet (Navigli and Ponzetto, 2010) can be seen as the cornerstone and the summit in that respect. However, if on the one hand WordNet's dense network of taxonomic relationships allows a high degree of systematization, on the other hand, a key unsolved issue with "*wordnets*" is the fine granularity of their inventories. Note that multilingualism in BabelNet is provided as an indexing service rather than as an alignment and unbiasing systematization method.

Extensions of these resources also include Common-Sense Knowledge (CSK), which refers to some (to a certain extent) widely-accepted and shared information. CSK describes the kind of general knowledge material that humans use to define, differentiate and reason about the conceptualizations they have in mind (Ruggeri et al., 2019). ConceptNet (Speer et al., 2017) is one of the largest CSK resources, collecting and automatically integrating data starting from the original MIT Open Mind Common Sense project[3]. However, terms in ConceptNet are not disambiguated. Property norms (McRae et al., 2005; Devereux et al., 2014) represent a similar kind of resource, which is more focused on the cognitive and perception-based aspects of word meaning. Norms, in contrast with ConceptNet, are based on semantic features empirically-constructed via questionnaires producing lexical (often ambiguous) labels associated with target concepts, without any systematic methodology of knowledge collection and encoding.

Another widespread modeling approach is based on vector space models of lexical knowledge. Vectors are automatically learnt from large corpora utilizing a wide range of statistical techniques, all centered on Harris' distributional assumption (Harris, 1954), i.e. words that occur in the same contexts tend to have similar meanings. Well-known models include word embeddings (Mikolov et al., 2013; Pennington et al., 2014; Bojanowski et al., 2016), sense embeddings (Huang et al., 2012; Iacobacci et al., 2015; Kumar et al., 2019), and contextualized embeddings (Scarlini et al., 2020). However, the relations holding between vector representations are not typed, nor are they organized systematically.

Among the several other modeling strategies proposed, lexicographic-centered resources have been focused on the contextualization of lexical items within syntactic structures, e.g. Corpus Pattern Analysis (CPA) (Hanks, 2004), situation frames such as FrameNet (Fillmore, 1977; Baker et al., 1998) and conceptual frames (Moerdijk et al., 2008; Leone et al., 2020). Words are not taken in isolation and the meaning they are attributed is connected to prototypical patterns or typed slots. However, these theories and methods for building semantic resources remain linked to the lexical basis and do not manage the mentioned biases.

[2]Words considered as synonyms in specific contexts.

[3]`https://www.media.mit.edu/projects/open-mind-common-sense/overview/`

3 The Multilingual Word Alignment

As is known, a single word form can be associated with more than one related sense, causing what is referred to as semantic ambiguity, or polysemy. This phenomenon, however, manifests itself differently across languages, since each language encodes meaning into words in its own particular way. We can therefore assume that, while a given polysemous word may be ambiguous in a certain context, a semantically corresponding word in another language will possibly not. Based on this assumption, it is possible to exploit this cross-language property to disambiguate a given word using its semantic equivalent in another language when they both occur in the same context. Such disambiguation process can take place because the two words feature different semantic - specifically, polysemous - behaviours. Accordingly, we developed a knowledge acquisition methodology that features the power of word sense disambiguation, relying on a multilingual $<$*target-concept*, $\{$*related words*$\}^k>$ alignment mechanism.

After providing a brief illustration of the languages we have selected for this first trial, we describe more in detail the methodology by using a basic example. Afterwards, a simple implementation of the proposed mechanism is presented.

3.1 Languages Involved

Among the benefits provided by the multilingual word alignment methodology we propose, one is that it prevents the represented lexical information from containing strong cultural-linguistic biases. This objective is pursued through the use of three different languages, reflecting in turn three diverse backgrounds. For this first trial we involved English, German and Italian. These languages were chosen primarily because we are proficient in them, therefore we are able to exert control over the data of our trial, as well as to interpret the results properly. Concurrently, given the nature of the methodology, it was necessary to select a set of languages with a certain degree of similarity in terms of shared lexical-semantic material. Indeed, the alignment mechanism can work and be effective as long as the lexical-semantic systems of the languages involved reflect a somewhat similar cultural-linguistic background. For example, we might expect languages to agree on the meanings of "carp", "cottage" and "sled" as long as speakers of these languages have comparable exposure to the relevant data. We would not expect a language spoken in a place without carps to have a word corresponding to "carp". The purpose of this project is not to forcibly identify universally valid semantic relationships, rather to not report biased information deriving from the use of data coming from a single linguistic context. For this reason, in our case the choice fell on European languages [4] (two Germanic languages and a Romance one).

wool	**Wolle**	**lana**
sheep	*Schal*	*cotone*
cotton	*spinnen*	*Biella*
synthetic	*Baumwolle*	*sintetica*
spin	*Rudolf*	*sciarpa*
scarf	*synthetisch*	*pecora*
mitten	*Schafe*	*filare*

Table 1: Unordered lists of single-language related words for $<$*wool* (EN), *Wolle* (DE), *lana* (IT)$>$.

3.2 Method

We now describe in detail the alignment mechanism through a basic example. Consider the following word forms: *wool* (EN); *Wolle* (DE); *lana* (IT), expressing a single target concept[5].

For each of the three lexical forms we collect a set of related words in terms of paradigmatic (e.g. synonyms) and syntagmatic (e.g. co-occurrences) relations. The target-related words can possibly be modifiers, verbs, or substantives. We thus obtain three different lists of words, one for each of the languages involved. The retrieved terms in the lists are still potentially ambiguous, since they refer to a lexical form rather than to a contextually defined concept. Table 1 provides a small excerpt of such unordered lists of related words.

The lexical data in the lists are subsequently compared and filtered in order to select only the semantic items that occur in all the lists, i.e., those shared by the three languages[6], in the reported example. The resulting words are thus aligned with their semantic counterparts, generating a set of aligned triplets, as shown in Table 2.

This multilingual word alignment provides, as a consequence, an automatic Word Sense Disambiguation system. Once the triplets are formed, their members will be indeed associated with a

[4]By "European" we refer to the European linguistic area.

[5]An absolute monosemy is, of course, realistically unreachable.

[6]This implies the presence of a translation step.

wool		Wolle		lana
sheep	↔	*Schafe*	↔	*pecora*
cotton	↔	*Baumwolle*	↔	*cotone*
synthetic	↔	*syntetisch*	↔	*sintetica*
spin	↔	*spinnen*	↔	*filare*
scarf	↔	*Schal*	↔	*sciarpa*

Table 2: Examples of aligned concept-related words for <*wool* (EN), *Wolle* (DE), *lana* (IT)>.

likely unique sense, i.e. the one coming from the intersection of all possible language-specific senses related to the three words. In other terms, the target-related words, once aligned, naturally identify (and provide) a common semantic context. As a consequence, potentially polysemous words are disambiguated through such context, without any support from sense repositories. For example, the context-consistent sense of the verb *to spin* (EN), which is a highly polysemous word in English, can be identified by selecting the only sense that is also shared by the other two aligned words, i.e. "*turn fibres into thread*". In fact, neither *spinnen* (DE) nor *filare* (IT) can possibly mean e.g. "rotate".

This mechanism generates a twofold effect: besides performing word sense disambiguation, it also provides lexical knowledge in the form of (paradigmatic and syntagmatic) lexical-semantic relations between words that is also language-unbounded. In the first place, the uncontrolled character of the data retrieval and alignment process offers the generation of novel lexical-semantic relations that are likely not available in other structured resources. Additionally, since the resulting set of words related to the target can be only the one shared by multiple languages, the lexical knowledge it encodes does not reflect a single cultural/linguistic background, rather a common and shared one. For example, in Table 1 the presence of the word "*Biella*" among the list of words related to "*lana*", probably refers to the fact that the Italian city Biella is (locally) famous for its wool, therefore the two words may co-occur frequently. Similarly, if we consider the alignment <*cat* (EN), *Katze* (DE), *gatto* (IT)>, a lexeme related to the English word form would be "*rain*", due to the well-known idiom "*it's raining cats and dogs*". However, neither "*Biella*" nor corresponding words for "*rain*" can possibly result in the lists of related words of the respective other languages, being language-specific items within those contexts. Therefore, the lexical information provided by the alignment mechanism will be free from strong cultural-linguistic biases. Finally, as illustrated in the next section, by exploiting multiple and differently built resources, we are able to reduce arbitrariness and lexicographic biases within the lexical knowledge represented.

4 Implementation

In this section we describe details and results of a simple implementation of the proposed alignment mechanism for the acquisition of disambiguated and unbiased lexical information. In particular, the system is composed of two main modules: a context generation and an alignment procedure. We finally report the results of an evaluation to highlight mainly *(i)* the autonomous disambiguation power of the approach, *(ii)* the quality of the alignments and their unbiased and syntagmatic nature, and *(iii)* the amount of unveiled lexical-semantic relations not covered by existing state-of-the-art resources such as BabelNet.

POS	scale	bilancia	Waage
noun	accuracy	precisione	Genauigkeit
noun	balance	equilibrio	Balance
noun	bulk	massa	Masse
noun	control	controllo	Kontrolle
noun	device	dispositivo	Gerät
noun	figure	cifra	Zahl
adj	accurate	preciso	genau
adj	smart	intelligente	intelligent
verb	indicate	indicare	zeigen
verb	set	regolare	einstellen

Table 3: 10 automatic alignments (out of 74) for the target concept <*scale* (EN), *bilancia* (IT), *Waage* (DE)> (BabelNet synset:00069470n).

4.1 Context for Multilingual Alignment

To retrieve the concept-related words for the multilingual alignment we made use of two textual resources: Sketch Engine (Kilgarriff et al., 2014) and the Leipzig Corpora Collection (Quasthoff et al., 2014). Through the former, we searched for related words with its tool named "Word Sketch" on the TenTen Corpus Family[7]. In particular, we were able to automatically collect words appearing in the following grammatical relations: "*mod-*

[7] https://www.sketchengine.eu/documentation/tenten-corpora

	00008050n	00069470n	00069470n	00062766n	00008364n	00008363n	
(en)	*libra*	*scale*	*plane*	*plane*	*bank*	*bank*	
(it)	*bilancia*	*bilancia*	*aereo*	*piano*	*banca*	*riva*	
(de)	*Waage*	*Waage*	*Flugzeug*	*Ebene*	*Bank*	*Ufer*	
triplets	**26**	**74**	**272**	**151**	**349**	**80**	
novel(en)	88,46%	87,84%	88,97%	89,40%	87,68%	91,25%	**88,9%**
novel(it)	76,92%	66,22%	75,74%	73,51%	75,64%	68,75%	**72,8%**
novel(de)	88,46%	74,32%	87,87%	84,11%	81,66%	76,25%	**82,1%**

Table 4: Alignments for six ambiguous concepts and percentage of unveiled *novel* relations in each language with respect to the BabelNet database. Some examples of triplets for the concept *scale-bilancia-Waage* (bn:*00069470n*) are shown in Table 3.

ifiers of w", "*adj. predicates of w*", "*verbs with w as subject*" and "*verbs with w as object*". The retrieved concept-related words are then lemmatized and marked with the suitable POS tags. Finally, we utilized the Leipzig Corpora Collection portal for searching additional context words in terms of left and right (POS-tagged) co-occurrences.

4.2 Multilingual Alignment

The Google Translate API was used for finding translations of related words in the three languages[8]. In particular, given a certain term t^{L1} in a language $L1$, we opted for retrieving all its possible translations into the other two languages ($L2$, $L3$). We then tried to match each translated item with the previously-retrieved sets of related words in $L2$, $L3$. Whenever the [$t^{L1} \leftrightarrow t^{L2}$]; [$t^{L1} \leftrightarrow t^{L3}$] match succeeded, we finally checked any possible [$t^{L2} \leftrightarrow t^{L3}$] match. If a [$t^{L1} \leftrightarrow t^{L2} \leftrightarrow t^{L3}$] semantic equivalence occurs, then the alignment can take place. Table 3 shows an excerpt of automatic alignments for the concept *scale* (bn:*00069470n*).

4.3 Evaluation

Our aim is not to overcome state-of-the-art resources but rather to incorporate new and unbiased semantic relations from a novel multilingual alignment mechanism. In particular, we wanted to verify to what extent our knowledge acquisition method is able to unveil lexical relations yet uncovered by a state-of-the-art resource (BabelNet).

Thus, we first generated sets of related words from BabelNet in order to compare them with those produced and aligned by our (automatized) methodology. In particular, through the BabelNet API, we obtained the English, Italian, and German lexicalizations of the synsets connected to it, together with the words included in their glosses[9].

As test cases, we randomly picked 500 concepts constituting polysemous words in at least one of the three languages, obtaining non-empty alignments for 456 of them. In Table 4 we report the results of the alignment on six concepts.

Despite its limitations, our first implementation of the proposed methodology was able to discover a total of 76,152 multilingual alignments over the 456 concepts, with (on average) more than 80% novel semantic relations with respect to what is currently encoded in BabelNet across the three languages. Still, the extracted data represent mostly unbiased and disambiguated knowledge, leading towards the construction of a new large-scale and multilingual prototypical lexical database.

5 Conclusions and Future Work

In this paper we proposed an original methodology for acquiring and encoding lexical knowledge through a novel yet simple mechanism of multilingual alignment. The aim was to represent varied, disambiguated, and language-unbounded lexical knowledge by minimizing strong linguistic and lexicographic biases. A simple implementation and experimentation on 456 concepts carried to unveil around 76K aligned lexical-semantic features, of which more than 80% resulted new when compared with a current state-of-the-art resource such as BabelNet. Future directions include the use of more languages and large-scale runs over thousands of main concepts (Bentivogli et al., 2004; Di Caro and Ruggeri, 2019; Camacho-Collados and Navigli, 2017).

[8] No surrounding syntactic context for the words to align was available for more advanced Machine Translation.

[9] We used the SpaCy library to analyze, extract and lemmatize the text - `https://spacy.io`.

References

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90.

Luisa Bentivogli, Pamela Forner, Bernardo Magnini, and Emanuele Pianta. 2004. Revising the wordnet domains hierarchy: semantics, coverage and balancing. In *Proceedings of the workshop on multilingual linguistic resources*, pages 94–101.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Jose Camacho-Collados and Roberto Navigli. 2017. Babeldomains: Large-scale domain labeling of lexical resources. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 223–228.

Barry J Devereux, Lorraine K Tyler, Jeroen Geertzen, and Billi Randall. 2014. The cslb concept property norms. *Behavior research methods*, 46(4):1119–1127.

Luigi Di Caro and Alice Ruggeri. 2019. Unveiling middle-level concepts through frequency trajectories and peaks analysis. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pages 1035–1042.

Charles J Fillmore. 1977. Scenes-and-frames semantics. *Linguistic structures processing*, 59:55–88.

Patrick Hanks. 2004. Corpus pattern analysis. In *Euralex Proceedings*, volume 1, pages 87–98.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Eric H Huang, Richard Socher, Christopher D Manning, and Andrew Y Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proc. of ACL*, pages 873–882.

Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. 2015. SensEmbed: learning sense embeddings for word and relational similarity. In *Proceedings of ACL*, pages 95–105.

Ferenc Kiefer. 1988. Linguistic, conceptual and encyclopedic knowledge: Some implications for lexicography. In T. Magay and J. Zigány, editors, *Proceedings of the 3rd EURALEX International Congress*, pages 1–10, Budapest, Hungary, sep. Akadémiai Kiadó.

Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, and Vít Suchomel. 2014. The sketch engine: Ten years on. *The Lexicography*, 1(1):7–36.

Sawan Kumar, Sharmistha Jat, Karan Saxena, and Partha Talukdar. 2019. Zero-shot word sense disambiguation using sense definition embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5670–5681.

Valentina Leone, Giovanni Siragusa, Luigi Di Caro, and Roberto Navigli. 2020. Building semantic grams of human knowledge. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2991–3000.

Ken McRae, George S Cree, Mark S Seidenberg, and Chris McNorgan. 2005. Semantic feature production norms for a large set of living and nonliving things. *Behav. r. m.*, 37(4):547–559.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Fons Moerdijk, Carole Tiberius, and Jan Niestadt. 2008. Accessing the anw dictionary. In *Proc. of the workshop on Cognitive Aspects of the Lexicon*, pages 18–24.

Jane Morris and Graeme Hirst. 2004. Non-classical lexical semantic relations. In *Proceedings of the Computational Lexical Semantics Workshop at HLT-NAACL 2004*, pages 46–51, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Roberto Navigli and Simone Paolo Ponzetto. 2010. BabelNet: Building a very large multilingual semantic network. In *Proc. of ACL*, pages 216–225. Association for Computational Linguistics.

Martha Palmer, Hoa Trang Dang, and Christiane Fellbaum. 2007. Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Nat.Lan.Eng.*, 13(02):137–163.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–43.

Uwe Quasthoff, Dirk Goldhahn, and Thomas Eckart. 2014. Building large resources for text mining: The leipzig corpora collection. In *Text Mining*, pages 3–24. Springer.

Alice Ruggeri, Luigi Di Caro, and Guido Boella. 2019. The role of common-sense knowledge in assessing semantic association. *Journal on Data Semantics*, 8(1):39–56.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020. SensEmBERT: Context-Enhanced Sense Embeddings for Multilingual Word Sense Disambiguation. In *Proceedings of the 34th Conference on Artificial Intelligence*. Association for the Advancement of Artificial Intelligence.

Robert Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*.

William A Woods. 1975. What's in a link: Foundations for semantic networks. In *Representation and understanding*, pages 35–82. Elsevier.

Michael Zock and Chris Biemann. 2020. Comparison of different lexical resources with respect to the tip-of-the-tongue problem. *Journal of Cognitive Science*, 21(2):193–252.

The Annotation of Liber Abbaci, a Domain-Specific Latin Resource

Francesco Grotto[1], **Rachele Sprugnoli**[2], **Margherita Fantoli**[3],
Maria Simi[4], **Flavio Massimiliano Cecchini**[2], **Marco Passarotti**[2]
1. Scuola Normale Superiore, Italy
2. Università Cattolica del Sacro Cuore, Italy
3. KU Leuven, Belgium
4. Università degli Studi di Pisa, Italy
francesco.grotto1@sns.it,
{rachele.sprugnoli,flavio.cecchini,marco.passarotti}@unicatt.it
margherita.fantoli@kuleuven.be, maria.simi@unipi.it

Abstract

The *Liber Abbaci* (13th century) is a milestone in the history of mathematics and accounting. Due to the late stage of Latin, its features and its very specialized content, it also represents a unique resource for scholars working on Latin corpora. In this paper we present the annotation and linking work carried out in the frame of the project *Fibonacci 1202-2021*. A gold-standard lemmatization and part-of-speech tagging allow us to elaborate some first observations on the linguistic and historical features of the text, and to link the text to the Lila Knowledge Base, that has as its goal to make distributed linguistic resources for Latin interoperable by following the principles of the Linked Data paradigm. Starting from this specific case, we discuss the importance of annotating and linking scientific and technical texts, in order to (a) compare and search them together with other (non-technical) Latin texts (b) train, apply and evaluate NLP resources on a non-standard variety of Latin. The paper also describes the fruitful interaction and coordination between NLP experts and traditional Latin scholars on a project requiring a large range of expertise.

1 Introduction

Latin texts have a wide diachronic and diatopic extension that corresponds to a similarly large diversity of the textual genres they represent. Besides literary ones, a huge amount of Latin texts of several different genres can be found spread all over Europe and beyond. An important textual genre is represented by scientific treaties, which in many cases are interesting not only for their contents, but also because of the technical terminology they feature.

This is precisely the case for the *Liber Abbaci* ‘the book of the abacus’ by Leonardo of Pisa (also known as Fibonacci). Written in the very first years of the 1200s, it is a book on arithmetic promoting a style of calculation based on Arabic numerals without aid of an abacus. *Fibonacci 1202-2021* is a project financed by the Tuscany Region and involving the University of Pisa and the Galilei Museum in Florence, following the publication of a critical edition of the *Liber Abbaci* by Enrico Giusti (Fibonacci, 2020). The goal of the project is to produce an enhanced digital edition of this work by leveraging advanced publishing tools and investigating the use of computational linguistics techniques in order to uncover the wealth of linguistic, scientific and historical information contained in the book.

Besides its scientific interest, the *Liber Abbaci* features a very peculiar lexicon, not often represented in the currently available (linguistically annotated) corpora for Latin. In order to fill this gap, in the context of the project *Fibonacci 1202-2021* we have started performing the linguistic annotation of the *Liber Abbaci*, beginning from part-of-speech (PoS) tagging and lemmatization of a specific chapter of the book, chosen for its linguistic and historical interest. The dataset is freely available online[1].

This paper describes the process of annotation of the *Liber Abbaci* and two applications of its

[1] http://dialogo.di.unipi.it/LiberAbbaci

results, namely (a) the evaluation of a number of trained models for PoS tagging and lemmatization for Latin in out-of-domain fashion and (b) the interlinking of the annotated chapter with other linguistic resources for Latin through the Lila Knowledge Base (KB)[2].

2 Related Work

The research area dealing with the creation of linguistic resources and Natural Language Processing (NLP) tools for ancient languages has seen a remarkable growth during the last decade (Sprugnoli and Passarotti, 2020). This has primarily concerned Latin and Ancient Greek as essential media to access and understand the so-called Classical heritage. In particular, several annotated corpora of Latin texts are currently available in digital format: they follow different guidelines and tagsets and feature different layers of linguistic annotation. This section wants to provide a (far from exhaustive) overview of such resources to show how the dataset presented in this paper stands with respect to the state of the art.

The LASLA corpus contains 2,500,000 semi-manually annotated tokens. It covers a large portion of the extant Classical Latin literature. It was started in 1961 by the LASLA research center at the Université de Liège[3] and is still being expanded[4]. The corpus is considered to be a gold standard, since the annotation of every token has been manually verified by a philologist. The linguistic information consists of lemmatization, morphological tagging, and an additional syntactic layer for verbs (Verkerk et al., 2020). Texts cover various literary genres (theater, poetry, prose) and have a chronological extension ranging from the comedies of Plautus to the texts of Suetonius and Pliny the Younger. Recent additions reach later stages of Latin literature [5], but include neither Medieval nor Neo-Latin works. Natural sciences and technical works are weakly represented in the corpus, the treatise *De Agri Cultura* 'on agriculture' by Cato and the recently added *Naturales Quaestiones* 'investigations about nature' by Seneca being the only examples.

The corpus of Latin Lemmatized Texts released by Thibault Clérice (Clérice, 2021a) is formed by 21,222,911 tokens (17,804,769 without punctuation marks) and includes a large set of Classical and Late Latin texts available in a a number of open access corpora[6]. Clérice's corpus covers a very ample chronological span (up until the 9th century) as well as different genres: from Classical literature (Horace, Ovid, etc.), to Christian religious texts and legal texts. The linguistic annotation consists of lemmatization and full morphological description of the tokens , produced automatically by applying the Pie Latin LASLA+ model 0.0.6 (Manjavacas et al., 2019), fine-tuned on ca. 1,500,000 tokens taken from the LASLA corpus (Clérice, 2021b), with very good results concerning lemmatization and PoS tagging[7]. However, results appear to be less good on unknown tokens[8]. This difference underlines the difficulty of using automatic annotation tools on texts with a very specialized language, surely not found in LASLA, as is the case for Fibonacci's *Liber Abbaci*.

As for syntactically annotated corpora, five treebanks are currently available for Latin. They are the *Index Thomisticus* Treebank (IT-TB) (Passarotti, 2019), the PROIEL treebank (Haug and Jøhndal, 2008; Eckhoff et al., 2018), the Latin Dependency Treebank by the Perseus Digital Library (part of the Ancient Greek and Latin Treebank) (Bamman and Crane, 2007), the Late Latin Charter Treebank (LLCT) (Cecchini et al., 2020b) and the UDante treebank (Cecchini et al., 2020a). The treebanks include texts of different genres (literary, historical, philosophical and documentary) and periods (from Classical to Medieval), but technical works are not represented.

3 Dataset Creation and Analysis

The *Liber Abbaci* is made up of more than 270,000 tokens and is divided into 15 chapters of varying length. The choice of starting our manual annotation from chapter VIII *de reperiendis pretiis mercium per maiorem guisam* 'on finding out the price of goods through the "greater means"' is due to the

[2] https://lila-erc.eu

[3] http://web.philo.ulg.ac.be/lasla/presentation-du-laboratoire/

[4] See http://web.philo.ulg.ac.be/lasla/textes-latins-traites/.

[5] Of which some are already available: see http://web.philo.ulg.ac.be/lasla/textes-latins-en-cours-de-traitement/.

[6] For the full list, see https://github.com/lascivaroma/latin-lemmatized-texts/tree/0.1.2.

[7] For lemmatization, accuracy: 0.9734 . For PoS tagging, accuracy: 0.9651 .

[8] For lemmatization, accuracy: 0.8716 . For PoS tagging, accuracy: 0.9232 .

peculiarity of its content. Here, Fibonacci treats many simple business negotiations using proportions and referring to many examples taken from the entire Mediterranean world. The examples concern weight and monetary systems as well as the main products bought and sold in the 13^{th} century. This means that the text is rich of terminology specific of the mathematical domain but also of trade and commerce. Chapter VIII is made up of 29,858 tokens (including punctuation marks), thus covering about 10% of the total length of the *Liber Abbaci*.

3.1 Data Annotation

The manual annotation of chapter VIII is carried out by a master's degree student in Classical languages, with excellent knowledge of Latin but without any previous expertise in either linguistic annotation or computational linguistics. The overall effort of the work amounts to a total of 227 hours, including: training sessions, study of the guidelines and of terminology related to measures, coins and trade in the Middle Ages (Marcinkowski, 2003; Martinori, 1915), the actual annotation, the reconciliation after evaluation of inter-annotator agreement (IAA, see Section 3.2), periodic checks with supervisors, the linking of the annotated text to the LiLa KB (see Section 5). We make use of a large number of dictionaries as references: the *Oxford Latin Dictionary* (OLD) (Souter, 1968), the *Lexicon Totius Latinitatis* (Forcellini, 1965), the *Dictionnaire illustré latin-français* (herafter: Gaffiot) (Gaffiot, 2016) and the *Thesaurus Linguae Latinae*[9] for Classical Latin, but also the *Dictionary of Medieval Latin from British Sources* (Latham and Howlett, 1975) and the *Glossarium mediae et infimae latinitatis* (du Cange et al., from 1883 to 1887) for Medieval Latin. Tokenization and sentence splitting are performed manually on a text editor, then lemmatization and PoS tagging are carried out on a shared spreadsheet following the Universal Dependencies (UD) formalism (de Marneffe et al., 2021), in particular both the universal and the language-specific guidelines relative to the latest release of the UD treebanks (v 2.9)[10].

The implementation of the UD guidelines to the linguistic peculiarities of the text does not always happen straightforwardly. Chapter VIII of the *Liber Abbaci*, as well as the work in its entirety, presents several typical features of Medieval Latin, both graphically (e. g. the monophthongization *ae* $\rightarrow$ *e* and the spelling *nichil* instead of the Classical *nihil* 'nothing'), morphologically (e. g the presence of analytical verb forms such as the "perfect", i. e. present perfective, subjunctive *habeat … honeratum*, instead of the Classical *onerauerit*, from *onero* 'to load') and syntactically (e. g. the nearly exclusive use of *quod* 'that' to introduce declarative clauses, instead of accusative and infinitive[11]). It is also worth noting the very limited use of enclitic particles (in the whole chapter VIII, Fibonacci uses the enclitic conjunction *que* 'and' only 3 times, appended to the auxiliary verb form *erunt* 'they will be') and the presence of syntactic calques of vernacular constructions (e. g. *secundum quod uadis multiplicando* 'according to what you are multiplying', where *uado* is preferred to the more Classical *eo* 'to go' and further assumes an auxiliary function, and the use of the gerundive form *multiplicando* is an innovation).

But the main peculiarities of the text concern the lexicon. Chapter VIII presents indeed a rich set of toponyms, units of measurement, names of coins and Arabisms often not even reported by Medieval Latin dictionaries. This is the case, for example, of some names of places, such as *Bugea*, today's Biğāya/Bgayet in Algeria (a city where Fibonacci spent a period of his childhood, learning the art of calculation), and *Septis*, today's Ceuta/Sabta on the Strait of Gibraltar; or, among the numismatic terms, of *bolsonalia*, a word designating a certain amount of broken silver or mixture coins which were sold to goldsmiths because they were adulterated or out of date.

3.2 Inter-Annotator Agreement

The IAA is calculated on 30 sentences (1,010 tokens), with the participation of a second scholar with a background in Classical languages. We register an almost perfect agreement with a Cohen's κ (Artstein and Poesio, 2008) of 0.97 for lemmatization and 0.94 for PoS tagging.

The comparison between the two annotations highlights two main issues. The first concerns the choice of the UPOS (Universal Part Of Speech) tag (de Marneffe et al., 2021, §2.2.2) for terms such as

[9] `https://thesaurus.badw.de/das-projekt.html`

[10] `https://universaldependencies.org/guidelines.html`

[11] See for example (Traina and Bertotti, 2015, C. XVI) .

nam ‘certainly’ and *enim* ‘namely’, because different corpora and dictionaries adopt different conventions: e. g. *nam* is labeled as `adverb` in the Lila KB and `Df` in the Latin PROIEL treebank, both possibly equivalent to UPOS `ADV`[12]; as `S`[13], standing for *conjonction de coordination* (UPOS: `CCONJ`) in the LASLA corpus, and more generically *conjonction (servant à confirmer/causale)* (UPOS either `CCONJ` or `SCONJ`) in the Gaffiot; finally *particle* (not necessarily corresponding to UPOS `PART`) in the OLD, and similarly *particule* in one sense in the Gaffiot. The treatment of the etymologically related and functionally similar *enim* is mostly identical for all sources, only with the Gaffiot reporting a sense as *adverbe* instead of *particule*, followed by the LASLA corpus in using both labels `S` and `M` (generic for *adverbe*), the latter though very marginally. These terms have been discussed and finally assigned the UPOS `PART`, used in the latest Latin UD treebanks to label discoursive particles like these. Such difficulties derive on one hand from the “volatile” and diachronically variable nature of similar elements, but on the other hand, and relatedly, to traditional grammars overlooking them and more generally skipping over pragmatic phenomena, in favour of “more Classical” parts of speech (hence the frequent inclusion of *nam*, *enim*, etc. in the catchall category of “adverbs”).

The second issue is the UPOS to be used for *unus* ‘one’. Fibonacci often uses *unus* to indicate a generic entity, as is clearly visible when paralleled by *alter* ‘other’. In this case, *unus* is tagged as `DET` (determiner), like *alter*[14]. In a number of other contexts, however, *unus* specifies the quantity of a certain object. In such cases it is considered a `NUM` (numeral)[15]. The difficulty here originates from a well known and complex linguistic change that will eventually produce a clear indefinite article from the numeral in Romance languages, but for which, being so gradual, we cannot pinpoint an exact historical moment; cf. (Ledgeway, 2012, §4.2.1).

4 Comparing NLP Models

Table 1 reports accuracy scores computed on our gold standard processed with UDPipe using the UD v2.6 models for Latin (Straka and Straková, 2017). The scores clearly show that current models are not good enough to process the Latin of Fibonacci. The best accuracy for lemmatization is achieved by the model trained on the LLCT treebank, which contains a set of Early Medieval charters written in Tuscany. However, this scores are lower than state-of-the-art ones: the best participating system at the EvaLatin 2020 evaluation campaign achieves an accuracy of $96,19\%$ for lemmatization and $96,74\%$ for PoS tagging on the corresponding test set (Sprugnoli et al., 2020), i. e. about 33 and 15 points more than the results obtained on Fibonacci.

	Lemma	UPOS
EvaLatin2020	63.60	81.90
IT-TB	65.58	77.14
LLCT	68.81	82.79
Perseus	67.54	78.37
PROIEL	60.25	51.64

Table 1: Accuracy of UDPipe v2.6 Latin models tested on chapter VIII of the *Liber Abbaci*.

Taking into consideration lemmatization, the percentage of out-of-vocabulary lemmas, that is, lemmas present in the text by Fibonacci but not in the training texts of the models, is very high ($> 50\%$ of lemma types). The majority of errors are registered for numbers and common nouns. The first problem is due to the fact that some models do not recognize Arabic numbers, because they have not seen them in their training data, while others lemmatize them with a special “metalemma” of the kind of *num. arab.*, eschewing lexical forms. As for common nouns, most errors related to lemmatization concern the lexical classes discussed in Section 3. For example, the tokens *libris* and *libre* are often lemmatized as *liber* ‘free’ (`ADJ`) instead of *libra* ‘pound’ (`NOUN`).

Table 2 shows the F1 score per UPOS tag. We observe that an F1 above 70% is achieved by any model only on 5 tags: `ADP`, `NOUN`, `NUM`, `SCONJ` and `VERB`. No model recognizes the `SYM` tag (used for mathematical operators such as paren-

[12] Cf. (Eckhoff et al., 2018, §5)

[13] With only very few exceptions when it is seen as part of a compound expression with tmesis, thus not receiving an autonomous PoS; cf. Pl. Am. 2.1, 49-50: ***Quo*** *id, malum, pacto potest* ***nam*** *(mecum argumentis puta) fieri, nunc uti tu et hic sis et domi?*, interpreted as an instance of *quonam* ‘whither pray?’, itself receiving `K` meaning *pronom interrogatif*.

[14] For instance, in the clause *ita est pretium* ***unius*** *ad pretium alterius* (VIII, 8) ‘so the price of **the one** [merchandise] is to the price of the other’.

[15] For instance, in the clause . . . *que multiplica per summam denariorum* ***unius*** *libre* (VIII, 20) ‘which you have to multiply by the amount of *denarii* of which **one** pound consists’.

	EvaLatin2020	IT-TB	Perseus	PROIEL	LLCT
SYM	0.00	0.00	0.00	0.00	0.00
AUX	0.24	0.45	0.03	0.27	0.32
ADJ	0.48	0.37	0.40	0.28	0.55
PRON	0.57	0.38	0.40	0.52	0.93
PART	0.65	0.00	0.00	0.00	0.65
ADV	0.70	0.71	0.78	0.21	0.84
CCONJ	0.75	0.67	0.68	0.44	0.86
SCONJ	0.89	0.95	0.96	0.86	0.95
VERB	0.91	0.87	0.92	0.78	0.84
NOUN	0.92	0.83	0.86	0.75	0.88
DET	0.93	0.00	0.00	0.53	0.91
PROPN	0.94	0.09	0.00	0.32	0.57
NUM	0.95	0.96	0.96	0.75	0.99
ADP	0.99	0.98	0.93	0.91	0.88
Global	0.71	0.52	0.49	0.46	0.73

Table 2: F1 on UPOS tags of UDPipe v2.6 Latin models on chapter VIII of the *Liber Abbaci*.

theses), because it is not present in their respective training data. The same is true for the tag PART in IT-TB (up until UD v2.8)[16], Perseus and PROIEL, and for the tag DET in Perseus. In old versions of the IT-TB, DET is limited to the proto-article *ly* (8 occurrences), while in Perseus the tag PROPN appears only for the lemma *Aefulanus* (1 occurrence). The IT-TB-based model, too, registers a near-zero F1 score for PROPN: in the corresponding training data, this tag is used for a restricted (116 types of lemmas) set of terms mostly specific to the domains of philosophy and religion (e. g. *Aristoteles*, *Maria*), not present in our dataset. Low performances are registered also for the AUX tag, the annotation of which is not consistent in training data: in Perseus, this tag is not used at all, while in EvaLatin 2020 it marks only the auxiliaries in periphrastic passive (including deponent) constructions, while in the other treebanks it is applied also to verbal copulas, as per UD guidelines. Further, the *Liber Abbaci* sees the rise (1 occurrence) of *habeo* 'to have' as a possible auxiliary (cf. Section 3.1), unheard of in Classical Latin and only attested (albeit marginally) in LLCT.

[16]Annotation discrepancies with respect to other Latin UD treebanks for INTJ, NUM, PART, PRON and DET have been resolved in IT-TB in its last version (2.9), released in November 2021; however, the model adopted in this paper and currently available in UDPipe is based on an older version of the data.

5 Linking and Querying in LiLa

The LiLa KB makes linguistic resources for Latin interoperable by linking tokens in corpora and entries in dictionaries/lexica to a collection of canonical forms for Latin called Lemma Bank (Passarotti et al., 2020). In order to connect the lemmas of chapter VIII to LiLa's KB, a string match is first performed between the lemmas in the texts and those in the KB, also taking into account their parts of speech. Using this strategy, 88.8% of the lemmas are directly connected to a single entry in the KB. The remaining unconnected lemmas fall into two possible categories: ambiguous lemmas, that is, with possible connection to more than one entry in the KB; and lemmas absent from the KB. More specifically, we find 44 ambiguous lemmas (corresponding to 631 tokens): for example, *colligo* can be connected to two entries: either a first-conjugation verb *colligare*[17] 'to bind', or a third-conjugation verb *colligĕre*[18] 'to gather'. These cases are manually disambiguated, checking each context of use. The remaining, not directly connected lemmas are not present in the KB and need to be manually added: these are mainly words denoting weight and monetary units (e. g. *karatus* 'carat'), or different written representations of lemmas already in LiLa (e. g. *torscellus* is a graphic variant of *tor-*

[17]https://lila-erc.eu/data/id/lemma/94854
[18]https://lila-erc.eu/data/id/lemma/94855

cellus[19], a unit of length). Thanks to the linking, each lemma of our dataset becomes part of an interoperable ecosystem made of resources of different kinds. We can thus query different interlinked resources using SPARQL and the LiLa endpoints[20]. For example, we can find the lemmas appearing only in chapter VIII[21] and not in the other texts that are currently linked to the KB: the *Summa Contra Gentiles* by Thomas Aquinas (from the *Index Thomisticus*), those found in UDante (a corpus of 5 works mostly by Dante Alighieri, or attributed to him, manually annotated following the UD formalism), and the *Querolus siue Aulularia* (an anonymous comedy dating back to the 5th c. AD).

Lemma	Gloss	Freq.
rotulus	unit of weight	296
soldus	monetary unit	212
virgula	bar of a fraction	202
byzantius	monetary unit	73
cantare	unit of weight	67

Table 3: The 5 most frequent distinctive lemmas in chapter VIII of the *Liber Abbaci*.

Table 3 shows the 5 most frequent distinctive, i. e. exclusively found in the *Liber Abbaci*, lemmas retrieved using a SPARQL query[22]. They are all related to mathematics, coins and units of measurement, confirming the specificity of the domain of our dataset. In particular, *rotulus* and *cantāre* are two units of weight, both deriving from Arabic, respectively from *raṭl* (in turn, a metathetical adaptation of Greek λίτρα *litra* 'pound') and *qinṭār*, which designates a weight of 100 *rotuli*[23]. The term *soldus*, instead, indicates a unit of measurement used for monetary quantities. Among the many currencies mentioned in chapter VIII, Fibonacci often cites the *byzantius*, a golden coin minted in Constantinople[24]. Finally, *virgula* (diminutive of *virga*, properly a 'rod', used by Fibonacci in the same sense of *virgula*) primarily denotes the bar between the numerator and denominator of a fraction, but it can also designate the fraction itself (Bocchi, 2004).

[19]https://lila-erc.eu/data/id/lemma/133810

[20]https://lila-erc.eu/sparql/

[21]https://lila-erc.eu/data/corpora/CorpusFibonacci/id/corpus/Liber Abbaci

[22]https://github.com/CIRCSE/SPARQL-queries/blob/main/distinctivelemmas-Fibonacci.rq

[23]It should be noted that Fibonacci alternates a third-declension *cantāre* (gen. sing. *cantāris*) with a second-declension *cantarium* (gen. sing. *cantarii*). During lemmatization of the text, the various attested singular forms have been linked to their respective lemmas; the nom./acc. plur. *cantaria*, which theoretically could derive both from *cantāre* and *cantarium*, has been linked to the lemma *cantāre* for simple reasons of probability, as it is the most frequently used by Fibonacci among these two forms.

6 Conclusions and Future Work

This paper describes the annotation of one chapter of the *Liber Abbaci* by Fibonacci, and reports on the linguistic peculiarities of this text and the ensuing challenges.

The results of existing UDPipe models in lemmatization and tagging show low accuracy and F1 scores when compared to the state of the art for these tasks in the recent EvaLatin 2020 evaluation campaign. This, on the one hand, can be attributed to the characteristics of the genre of Fibonacci's texts, which are representative of scientific Medieval Latin texts, and on the other hand can be explained with the different choices in annotation style of Latin treebanks released under the UD project. Substantial improvements can be expected with models trained on new releases of Latin treebanks which have already undertaken the effort of resolving annotation discrepancies and of making the annotation style across treebanks more homogeneous. Further improvements will however require new annotated chapters and experiments in domain adaptation, which are scheduled as future work.

Acknowledgments

This work is a contribution to the *Fibonacci 1202-2021* project, financed by the Tuscany Region. Part of the work has been funded by the European Research Council (ERC) under the European Union's *Horizon 2020* research and innovation programme – Grant Agreement No. 769994. The authors want to thank: prof. Andrea Bocchi, dott. Alessandro Gelsumini, prof. Pier Daniele Napolitani and prof. Enrica Salvatori for their linguistic and historical advice.

References

Ron Artstein and Massimo Poesio. 2008. Inter-coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596.

[24]Also mentioned is the *byzantius saracenatus*, equivalent to the *hyperperus*, that is, a *byzantius* with inscriptions in Kufic characters (Martinori, 1915).

David Bamman and Gregory Crane. 2007. The Latin Dependency Treebank in a Cultural Heritage Digital Library. In *Proceedings of the Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2007)*, pages 33–40, Prague, Czech Republic, June. Association for Computational Linguistics.

Andrea Bocchi. 2004. In Michelangelo Zaccarello and Lorenzo Tomasin, editors, *Storia della lingua e filologia. Per Alfredo Stussi nel suo sessantacinquesimo compleanno*, chapter Sì nel Livero de l'abbecho, pages 121–158. SISMEL – Edizioni del Galluzzo, Florence, Italy.

Flavio M. Cecchini, Rachele Sprugnoli, Giovanni Moretti, and Marco Passarotti. 2020a. UDante: First Steps Towards the Universal Dependencies Treebank of Dante's Latin Works. In *Seventh Italian Conference on Computational Linguistics*, pages 1–7, Bologna. CEUR-WS.org.

Flavio Massimiliano Cecchini, Timo Korkiakangas, and Marco Passarotti. 2020b. A New Latin Treebank for Universal Dependencies: Charters between Ancient Latin and Romance Languages. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 933–942, Marseille, France, May. European Language Resources Association.

Thibault Clérice. 2021a. lascivaroma/latin-lemmatized-texts: 0.1.2 - HN PSL, May. DOI: 10.5281/zenodo.4661034; project online at https://github.com/lascivaroma/latin-lemmatized-texts.

Thibault Clérice. 2021b. Latin Lasla Model, Apr. DOI: 10.5281/zenodo.4661034.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308, 07.

Charles du Fresne sieur du Cange, bénédictins de la congrégation de Saint-Maur, d. Pierre Carpentier, Johann Christoph Adelung, G. A. Louis Henschel, Lorenz Diefenbach, and Léopold Favre. from 1883 to 1887. *Glossarium mediae et infimae latinitatis*. Favre, Niort, France.

Hanne Martine Eckhoff, Kristin Bech, Gerlof Bouma, Kristine Eide, Dag Haug, Odd Einar Haugen, and Marius Jøhndal. 2018. The PROIEL treebank family: a standard for early attestations of Indo-European languages. *Language Resources and Evaluation*, 52(1):29–65.

Leonardus Bigollus Pisanus vulgo Fibonacci. 2020. *Liber Abbaci*, volume 79 of *Biblioteca di «Nuncius»*. Leo S. Olschki, Florence, Italy.

Egidio Forcellini. 1965. *Lexicon totius latinitatis*. Arnaldo Forni, Bologna, Italy.

Félix Gaffiot. 2016. *Dictionnaire Latin-Français*. Accessible at gaffiot.fr.

Dag Trygve Truslew Haug and Marius Jøhndal. 2008. Creating a Parallel Treebank of the Old Indo-European Bible Translations. In *Proceedings of the Second Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2008)*, pages 27–34.

Ronald Edward Latham and David R Howlett. 1975. *Dictionary of Medieval Latin from British Sources: Fascicule V: IJKL*. OUP Oxford.

Adam Ledgeway. 2012. *From Latin to Romance*, volume 1 of *Oxford studies in historical and diachronic linguistics*. Oxford University Press, Oxford, UK.

Enrique Manjavacas, Ákos Kádár, and Mike Kestemont. 2019. Improving lemmatization of non-standard languages with joint learning. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1493–1503, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Hinz Marcinkowski. 2003. *Measures and Weights in the Islamic World, an English Translation of Walther Hinz's Handbook Islamische Maße und Gewichte*. International Islamic University Malaysia (IIUM).

Edoardo Martinori. 1915. *La Moneta: vocabolario generale*. Instituto italiano di numismatica.

Marco Passarotti, Francesco Mambrini, Greta Franzini, Flavio Massimiliano Cecchini, Eleonora Litta, Giovanni Moretti, Paolo Ruffolo, and Rachele Sprugnoli. 2020. Interlinking through lemmas. the lexical collection of the lila knowledge base of linguistic resources for latin. *Studi e Saggi Linguistici*, 58(1):177–212.

Marco Passarotti, 2019. volume 10 of *Age of Access? Grundfragen der Informationsgesellschaft*, chapter The Project of the Index Thomisticus Treebank, pages 299–320. De Gruyter Saur, Berlin, Germany; Boston, MA, USA.

Alexander Souter. 1968. *Oxford Latin dictionary: OLD*. Clarendon Press.

Rachele Sprugnoli and Marco Passarotti. 2020. Proceedings of LT4HALA 2020-1st Workshop on Language Technologies for Historical and Ancient Languages. In *Proceedings of LT4HALA 2020-1st Workshop on Language Technologies for Historical and Ancient Languages*.

Rachele Sprugnoli, Marco Passarotti, Flavio Massimiliano Cecchini, and Matteo Pellegrini. 2020. Overview of the EvaLatin 2020 evaluation campaign. In *Proceedings of LT4HALA 2020 - 1st Workshop on Language Technologies for Historical and Ancient Languages*, pages 105–110, Marseille, France, May. European Language Resources Association (ELRA).

Milan Straka and Jana Straková. 2017. Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada, August. Association for Computational Linguistics.

Alfonso Traina and Tullio Bertotti. 2015. *Sintassi normativa della lingua latina*. Pàtron, Bologna, Italy.

Philippe Verkerk, Yves Ouvrard, Margherita Fantoli, and Dominique Longrée. 2020. L.A.S.L.A. and Collatinus: a convergence in lexica. *SSL*, 1(LVIII):95–120.

Detecting Age-Related Linguistic Patterns in Dialogue: Toward Adaptive Conversational Systems

Lennert Jansen[1], **Arabella Sinclair**[1], **Margot J. van der Goot**[2], **Raquel Fernández**[1], **Sandro Pezzelle**[1]

[1]Institute for Logic, Language and Computation (ILLC), University of Amsterdam
[2]Amsterdam School of Communication Research (ASCoR), University of Amsterdam
lennertjansen95@gmail.com
{a.j.sinclair|m.j.vandergoot|raquel.fernandez|s.pezzelle}@uva.nl

Abstract

This work explores an important dimension of variation in the language used by dialogue participants: their age. While previous work showed differences at various linguistic levels between age groups when experimenting with written *discourse* data (e.g., blog posts), previous work on *dialogue* has largely been limited to acoustic information related to voice and prosody. Detecting fine-grained linguistic properties of human dialogues is of crucial importance for developing AI-based conversational systems which are able to adapt to their human interlocutors. We therefore investigate whether, and to what extent, current text-based NLP models can detect such linguistic differences, and what the features driving their predictions are. We show that models achieve a fairly good performance on age-group prediction, though the task appears to be more challenging compared to discourse. Through in-depth analysis of the best models' errors and the most predictive cues, we show that, in dialogue, differences among age groups mostly concern stylistic and lexical choices. We believe these findings can inform future work on developing controlled generation models for adaptive conversational systems.

age 19-29	
A: oh that's cool	
	B: different sights and stuff
A: oh	

age 50+	
A: well quite and I'd have to come back as well	
	B: that's of course
A: and make up for you know	

Figure 1: Example dialogue snippets from speakers of different age groups in the British National Corpus. We conjecture that stylistic and lexical differences between age groups can be detected. Here, we experiment at the level of the utterance.

1 Introduction

Research on developing conversational agents has experienced impressive progress, particularly in recent years (McTear, 2020). However, artificial systems that can tune their language to that of a particular individual or group of users continue to pose more of a challenge. Recent examples of this line of research include adaptation at style level (Ficler and Goldberg, 2017), persona-specific traits (Zhang et al., 2018), or other traits such as sentiment (Dathathri et al., 2020).

Personalised interaction is of crucial importance to obtain systems that can be trusted by users and perceived as natural (van der Goot and Pilgrim, 2019), but most of all to be accessible to varying user profiles, rather than targeted at one particular user group (Zheng et al., 2019; Zeng et al., 2020).

In this work, we focus on one particular aspect that may influence conversational agent success: user age profile. We investigate whether the linguistic behaviour of conversational participants differs across age groups using state-of-the-art NLP models on purely textual data, without considering vocal cues. We aim to detect age from characteristics of language use and adapt to this signal, rather than work from ground-truth metadata about user demographics. This is in the interest of preserving privacy, and from the perspective that while age and language use may have a relationship, this will not be linear (Pennebaker and Stone, 2003) and there are individual differences.

Previous work on age detection in dialogue has focused on speech features, which are known to systematically vary across age groups. For example, Wolters et al. (2009) learn logistic regression age classifiers from a small dialogue dataset using different acoustic cues supplemented with a small set of hand-crafted lexical features, while Li et al. (2013) develop SVM classifiers using acoustic and prosodic features extracted from scripted utterances spoken by participants interacting with an artificial system. In contrast to this line of work, we investigate whether different age groups can be detected from textual linguistic information rather than voice-related cues. We explore whether, and to what extent, various state-of-the-art NLP models are able to capture such differences in dialogue data as a preliminary step to age-group adaptation by conversational agents.

We build on the work of Schler et al. (2006), who focus on age detection in written discourse using a corpus of blog posts. The authors learn a Multi-Class Real Winnow classifier leveraging a set of pre-determined style- and content-based features, including part-of-speech categories, function words, and the 1000 unigrams with the highest information gain in the training set. They find that content features (lexical unigrams) yield higher accuracy (74%) than style features (72%), while their best results (76.2%) are obtained with their combination. We extend this investigation in several key ways: (1) we leverage state-of-the-art NLP models that allow us to learn representations end-to-end, without the need to specify concrete features in advance; (2) we apply this approach to dialogue data, using a large-scale dataset of transcribed, spontaneous open-domain dialogues, and also use this approach to replicate the experiments of Schler et al. (2006) on disccourse; (3) we show that text-based models can indeed detect age-related differences, even in the case of very sparse signal at the level of dialogue utterances; and finally (4) we carry out an in-depth analysis of the models' predictions to gain insight on which elements of language use are most informative.[1]

Our work can be considered a first step toward the modeling of age-related linguistic adaptation by AI conversational systems. In particular, our results can inform future work on controlled text generation for dialogue agents (Dathathri et al., 2020; Madotto et al., 2020).

[1]Code and data available at: `https://github.com/lennertjansen/detecting-age-in-dialogue`

age	**#samples**	**#tokens**	**mean L (± sd)**	**min-max L**
19-29	33,641	381,195	11.3 (±15.98)	1-423
50+	33,641	406,157	12.1 (±21.62)	1-1246
all	67,282	787,352	11.7 (±19.0)	1-1246

Table 1: Descriptive statistics of the dataset. L means length, i.e., number of tokens in a sample.

2 Data

We use a dataset of dialogue data where information about the age of the speakers involved in the conversation is available (see the dialogue snippets in Figure 1), i.e., the spoken partition of the British National Corpus (Love et al., 2017). This partition includes spoken informal open-domain conversations between people that were collected between 2012 and 2016 via crowd-sourcing, and then recorded and transcribed by the creators. Dialogues can be between two or more interlocutors, and are annotated along several dimensions including age and gender together with geographic and social indicators. Speaker ages are categorized in ten brackets: 0-10, 11-18, 19-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, and 90-99.

We focus on conversations that took place between two interlocutors, and only consider dialogues between people of the same age group. We then restrict our investigation to a binary opposition: *younger* vs. *older* age group. We split the dialogues into their constituent utterances (e.g., from each dialogue snippet in Figure 1 we extract three utterances), and further pre-process them by removing non-alphabetical characters. Only samples which are not empty after pre-processing are kept. For the *younger* group, we consider the 19-29 bracket, which contains 138,662 utterances. For the *older*, we group conversations from five brackets: 50-59, 60-69, 70-79, 80-89, and 90-99 (hence, 50+), which sums up to a total of 33,641 utterances. The choice of grouping these brackets is a trade-off between experimenting with fairly distinct age groups (the age difference between them is at least 20 years) and obtaining large-enough data for each of them.

We randomly sample 33,614 utterances from the 19-29 group in order to experiment with a balanced number of samples per group. The resulting dataset, that we use for our experiments, includes around 67K utterances with an average length of 11.7 tokens. Descriptive statistics are in Table 1.

3 Method

We frame the problem as a binary classification task: given some text, we seek to predict whether the age class of its speaker is *younger* or *older*.

3.1 Models

We experiment with various models, that we briefly describe below. Details on model training and evaluation are given at the end of the section.

***n*-gram** Our simplest models are based on n-grams, which have the advantage of being highly interpretable. Each data entry (i.e., a dialogue utterance) is split into chunks of all possible contiguous sequences of n tokens. The resulting vectorized features are used by a logistic regression model to estimate the odds of a text sample belonging to a certain age group. We experiment with unigram, bigram and trigram models. A bigram model uses unigrams and bigrams, and a trigram model unigrams, bigrams, and trigrams.

LSTM and BiLSTM We use a standard Long Short-Term Memory network (LSTM) (Hochreiter and Schmidhuber, 1997) with two layers, embedding size 512, and hidden layer size 1024. Batch-wise padding is applied to variable length sequences. The original model's bidirectional extension, the bidirectional LSTM (BiLSTM) (Schuster and Paliwal, 1997), is also used. Padding is similarly applied to this model, and the following optimal architecture is experimentally found: embedding size 64, 2 layers, and hidden layer size 512. Both RNN models are found to perform optimally for a learning rate of 10^{-3}.

BERT We experiment with a Transformer-based model, i.e., BERT (Devlin et al., 2019). BERT is pre-trained to learn deeply bidirectional language representations from massive amounts of unlabeled textual data. We experiment with the base, uncased version of BERT, in two settings: by using its pre-trained frozen embeddings (BERT$_{frozen}$) and by fine-tuning the embeddings on our age classification task (BERT$_{FT}$). BERT embeddings are followed by dropout with probability 0.1 and a linear layer with input size 768.

Experimental details The dataset is randomly split into a training (75%), validation (15%), and test (10%) set. Each model with a given configuration of hyperparameters is run 5 times with different random initializations. All models are trained on an NVIDIA TitanRTX GPU.

The n-gram models are trained in a One-vs-Rest (OvR) fashion, and optimized using the Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm (Liu and Nocedal, 1989), with a maximum of 10^6 iterations. The n-gram models are trained until convergence or for the maximum number of iterations.

LSTMs and BERT models are optimized using Adam (Kingma and Ba, 2015), and trained for 10 epochs, with an early stopping patience of 3 epochs. The RNN-based models' embeddings are jointly trained, and optimal hyperparameters (i.e., learning rate, embedding size, hidden layer size, and number of layers) are determined using the validation set and a guided grid-search. BERT$_{FT}$ is fine-tuned on the validation set for 10 epochs, or until the early stopping criterion is met. BERT has a maximum input length of 512 tokens. Sequences exceeding this length are truncated.

Model	**Accuracy** ↑ better	$\boldsymbol{F}_1^{(19-29)}$ ↑ better	$\boldsymbol{F}_1^{(50+)}$ ↑ better
Random	0.500	0.500	0.500
unigram	0.701 (0.007)	0.708 (0.009)	0.693 (0.004)
bigram	0.719 (0.002)	0.724 (0.003)	0.714 (0.003)
trigram	0.722 (0.001)	0.727 (0.003)	0.717 (0.001)
LSTM	0.693 (0.003)	0.696 (0.005)	0.691 (0.007)
BiLSTM	0.691 (0.009)	0.702 (0.017)	0.679 (0.007)
BERT$_{frozen}$	0.675 (0.003)	0.677 (0.008)	0.673 (0.010)
BERT$_{FT}$	**0.729** (0.002)	**0.730** (0.011)	**0.727** (0.010)

Table 2: Test set results averaged over 5 random initializations. Format: *average metric (standard error)*. Values in **bold** are the highest in the column; in blue, the second highest.

4 Results

We report accuracy and F_1 for each age group in Table 2. As can be seen, the performance of all models is well beyond chance level, which indicates that age-related linguistic differences can be detected, to some extent, even by a simple model based on unigrams. At the same time, BERT fine-tuned on the task turns out to be the best-performing model both in terms of accuracy (0.729) and F_1 scores, which confirms the effectiveness of Transformer-based representations to encode fine-grained linguistic differences. However, it can be noted that the model based on tri-

	% cases	avg. length (±std)*
both correct	63.17%	13.51 (±18.98)
both wrong	19.78%	5.82 (±8.33)
only trigram correct	7.91%	10.44 (±11.66)
only BERT correct	9.14%	11.53 (±12.12)

Table 3: Percentage cases of (non-)overlapping (in)correctly predicted cases between trigram and BERT$_{FT}$. *Utterance length measured in tokens.

grams is basically on par with BERT in terms of accuracy (0.722), and well above both the LSTM and BiLSTM models (0.693 and 0.691, respectively). A similar pattern is observed for F_1 scores, where BERT$_{FT}$ and the trigram model achieve comparable performance, with LSTMs being overall behind.

Overall, our results indicate that text-based models are effective, to some extent, in predicting the age group to which a speaker involved in a dialogue belongs. This complements previous evidence that age-related features can be detected in discourse (Schler et al., 2006), and shows that in dialogue the task appears to be somehow more challenging: The improvement in accuracy with respect to the majority/random baseline is lower in our dialogue results (+22.9%) as compared to what observed in discourse both by Schler et al. (2006) (+32.4%) and by us (+27%) when replicating their study using the models and experimental setup described in Section 3.1. Similarly to dialogue, BERT$_{FT}$ achieves the highest results in discourse (0.742). In contrast, both LSTMs (0.663) and n-grams (0.625) significantly lag behind it. Note that, although based on the same corpus of texts, i.e., the Blog Authorship Corpus,[2] and the same 3 age groups, i.e., 13-17, 23-27, and 33+, our replicated results are not fully comparable to those by Schler et al. (2006). Due to our more cautious data pre-processing, we experiment with more samples than they do (677K vs. 511K), which in turn leads to a different majority baseline.

There can be several reasons why age group detection is more challenging in dialogue than in discourse. For example, in dialogue there may be dimensions of variation, such as turn-taking patterns, that are not captured by our models and experimental setup. Yet, the present results do reveal a few interesting insights. In particular, the very good performance of the trigram model suggests that leveraging 'local' linguistic features captured by n-grams is extremely effective in *dialogue*. This could indicate that differences among various age groups are at the level of local lexical constructions. This deserves further analysis, that we carry out in the next section.

5 Analysis

We compare the two best-performing models, i.e., BERT$_{FT}$ and the one using trigrams, and aim to shed light on what cues they use to solve the task. We first compare the prediction patterns of the two models, which allows us to detect easy and hard examples. Second, we focus on the trigram model and report the n-grams that turn out to be most informative to distinguish between age groups.

5.1 Comparing Model Predictions

We split the data for analysis by whether or not both models make the same correct or incorrect prediction, or whether they differ. Table 3 shows the breakdown of these results. As can be seen, a quite large fraction of samples are correctly classified by both models (63.17%), while in 19.78% cases neither of the models make a correct prediction. The remaining cases are almost evenly split between cases where only one of the two is correct. As shown in Figure 2, the 19-29 age group appears to be be slightly easier compared to the 50+ group, where models make more errors.

To qualitatively inspect what the utterances falling into these classes look like, in Table 4 we show a few cherry-picked cases for each age group. We notice that, not surprisingly, both models have trouble with backchanneling utterances consisting of a single word, such as *yeah*, *mm*, or *really?*, which are used by both age groups. For example, both models seem to consider *yeah* as a 'young' cue, which leads to wrong predictions when *yeah* is used by a speaker in the 50+ group. As for the utterance *really?*, BERT$_{FT}$ assigns it to the 50+ group, while the trigram model makes the opposite prediction. This indicates that certain utterances simply do not contain sufficient distinguishing information, and model predictions that are based on them should therefore not be considered reliable. This seems to be particularly the case for short utterances. Indeed, through comparing the average length of the utterances incorrectly classified by both models (rightmost column

[2]The corpus contains blog posts appeared on `https://www.blogger.com`, gathered in or before August 2004.

age	both correct	both wrong	only BERT$_{FT}$ correct	only trigram correct
19-29	I don't know? sounded crazy	that's a lot of people for one house	yeah okay	really?
19-29	yeah	well there you go	oh I'm not very good at that	I've got a pen I've got a pen
19-29	do you have exams again?	mm	empty promises isn't it?	day of death and ice-cream
50+	and as I say	yeah	really?	well if I were you
50+	yes	that would be controversial	yeah it seems to	that's it
50+	oh really?	he's got that already	that we caused it	oh I thought you said Godzilla

Table 4: Examples where both models are correct/wrong or only BERT$_{FT}$/trigram is correct.

of Table 3), we notice that they are much shorter than those belonging to the other cases. This is interesting, and indicates a key challenge in the analysis of dialogue data: on average, shorter utterances contain less signal. On the other hand, short utterances can provide rich conversational signal in dialogue; for example, backchanneling, exclamations, or other acknowledging acts. As a consequence, using length alone as a filter is not an appropriate approach, as it can remove aspects of language use key to differentiating speaker groups.

5.2 Most Informative N-grams

Analyzing the most informative n-grams used by the trigram model allows us to qualitatively compare the linguistic differences inherent to each age group. In Table 5 we report the top 15 n-grams per group. We find, firstly and intuitively, that colloquial language seems somewhat generational, with unigrams particularly indicative of younger speakers consisting of words such as *cool* and *massive*, and for older speakers, words like *wonderful*. These unigrams are both informative to the model and indicative of differences in both formality and 'slang' use across age groups.

These most informative n-grams also indicate differences in back-channeling use between age groups; younger speaker's language is more characterized by the use of *um*, *hmm*, while the top

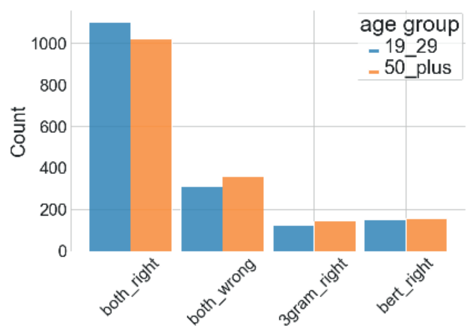


Figure 2: Distribution of predicted cases by trigram and BERT$_{FT}$ models, split by age groups.

n-grams in the older category will more likely use *yes*, *right*, *right right*. A feature of younger language also apparent from these examples is in their use of more informal language, which also extends to the use of foul language, making up a percent of the most informative unigrams shown in Table 5.

Interestingly, while topic words make up many of the most informative n-grams for older speakers in Table 5, younger speakers are more defined by their use of slang words such as *wanna*, foul language, or adjectives such as *cute*, *cool*, and *massive*. A key finding from Schler et al. (2006) is in the sentiment of language playing an important role, something which some of the most informative n-grams suggest may also be true for the dialogue dataset. As Table 5 demonstrates, younger speakers use more dramatic language such as negative foul words, and positive *love, cute, cool*; all words with a strong connotative meaning. We believe that further inspection is needed to determine whether the same sentiment pattern will be true of

	19-29		50+
coef.	**n-gram**	**coef.**	**n-gram**
-3.20	um	2.37	yes
-2.84	cool	2.12	you know
-2.58	s**t	2.09	wonderful
-2.12	hmm	1.90	how weird
-2.09	like	1.84	chinese
-2.02	was like	1.73	right
-1.96	love	1.71	building
-1.96	as well	1.66	right right
-1.88	as in	1.55	so erm
-1.84	cute	1.43	mm mm
-1.82	uni	1.41	cheers
-1.79	massive	1.39	shed
-1.79	wanna	1.37	pain
-1.79	f**k	1.36	we know
-1.72	tut	1.08	yeah exactly

Table 5: Top 15 most informative n-grams per age group used by the trigram model. **coef.** is the coefficient (and sign) of the corresponding n-gram for the logistic regression model: the higher its absolute value, the higher the utterance's odds to belong to one age group. * indicates foul language.

dialogue as it has been reported to be in discourse.

6 Conclusion

We investigated whether, and to what extent, NLP models can detect age-related linguistic features in dialogue data. We showed that, in line with what we observed for discourse, state-of-the-art models are capable of doing so with a reasonable accuracy, in particular when the dialogue fragment is long enough to contain discriminative signal. At the same time, we found that much simpler models based on n-grams achieve comparable performance, which suggests that, in dialogue, 'local' features can be indicative of the language of speakers from different age groups. We showed this to be the case, with both lexical and stylistic cues being informative to these models in this task.

While we performed the classification task at the level of single dialogue utterances, future work may take into account larger dialogue fragments, such as the entire dialogue or a fixed number of turns. This would make the setup more comparable to discourse, but would require making experimental choices and dealing with extra computational challenges. Moreover, it could be tested whether the language used by a speaker is equally discriminative when talking to a same-age (this work) or a different-age interlocutor.

Finally, we believe our findings could inform future work on developing adaptive conversational systems. Since consistent language style differences were found between age groups (for example, at the level of exclamatives and acknowledgments), systems whose language generation capabilities aim to be consistent with a given age group should therefore reproduce these patterns. This could be achieved, for example, by embedding one or more discriminative modules that control the generation of a system's output, which could lead to better, more natural interactions between human speakers and a conversational system.

Acknowledgements

This work received funding from the University of Amsterdam's Research Priority Area *Human(e) AI* and from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819455).

References

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation*, pages 94–104, Copenhagen, Denmark, September. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Ming Li, Kyu J Han, and Shrikanth Narayanan. 2013. Automatic speaker age and gender recognition using acoustic and prosodic level information fusion. *Computer Speech & Language*, 27(1):151–167.

Dong C Liu and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. *Mathematical programming*, 45(1):503–528.

R Love, C Dembry, A Hardie, V Brezina, and T McEnery. 2017. The Spoken BNC2014: designing and building a spoken corpus of everyday conversations. *In International Journal of Corpus Linguistics*, 22(3):319–344.

Andrea Madotto, Etsuko Ishii, Zhaojiang Lin, Sumanth Dathathri, and Pascale Fung. 2020. Plug-and-play conversational models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2422–2433, Online, November. Association for Computational Linguistics.

Michael McTear. 2020. Conversational AI: Dialogue systems, conversational agents, and chatbots. *Synthesis Lectures on Human Language Technologies*, 13(3):1–251.

James W Pennebaker and Lori D Stone. 2003. Words of wisdom: Language use over the life span. *Journal of Personality and Social Psychology*, 85(2):291–301.

Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. 2006. Effects of age and gender on blogging. In *AAAI spring symposium: Computational approaches to analyzing weblogs*, volume 6, pages 199–205.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

Margot J van der Goot and Tyler Pilgrim. 2019. Exploring age differences in motivations for and acceptance of chatbot communication in a customer service context. In *International Workshop on Chatbot Research and Design*, pages 173–186. Springer.

Maria Wolters, Ravichander Vipperla, and Steve Renals. 2009. Age recognition for spoken dialogue systems: Do we need it? In *Tenth Annual Conference of the International Speech Communication Association (Interspeech)*.

Guangtao Zeng, Wenmian Yang, Zeqian Ju, Yue Yang, Sicheng Wang, Ruisi Zhang, Meng Zhou, Jiaqi Zeng, Xiangyu Dong, Ruoyu Zhang, Hongchao Fang, Penghui Zhu, Shu Chen, and Pengtao Xie. 2020. MedDialog: Large-scale medical dialogue datasets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9241–9250, Online, November. Association for Computational Linguistics.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213, Melbourne, Australia, July. Association for Computational Linguistics.

Yinhe Zheng, Guanyi Chen, Minlie Huang, Song Liu, and Xuan Zhu. 2019. Personalized dialogue generation with diversified traits. *CoRR*, abs/1901.09672.

From Cambridge to Pisa: A Journey into Cross-Lingual Dialogue Domain Adaptation for Conversational Agents

Tiziano Labruna[1,2], **Bernardo Magnini**[1]
1. Fondazione Bruno Kessler, Italy
2. Free University of Bozen-Bolzano, Italy
tlabruna@fbk.eu, magnini@fbk.eu

Abstract

English. Domain and language shift are still major bottlenecks for a vast range of task-oriented dialogue systems. This paper focuses on data-driven models for dialogue state tracking, and builds on top of recent work on *dialogue domain adaptation*, showing that state-of-the-art models are very sensible to language shift obtained through automatic translation. Experiments show that combining training data for the two languages (English and Italian) is always beneficial, while combining domains does not increase performance. As a relevant side effect of our work, we present a new dataset for dialogue state tracking available for Italian, derived from MultiWOZ 2.3.

Italiano. *I cambiamenti di dominio e di lingua sono ancora uno dei maggiori ostacoli per una ampia classe di sistemi di dialogo task-oriented. Questo lavoro si focalizza su modelli derivati da dati per tracciare gli stati del dialogo, e prosegue lavori recenti su adattamento del dialogo al dominio, mostrando che i modelli allo stato dell'arte sono molto sensibili ai cambiamenti di lingua ottenuti tramite traduzione automatica. Gli esperimenti mostrano che combinando i dati di addestranmento per due lingue (inglese e italiano) e' sempre vantaggioso, mentre la combinazione di dominii non migliora le prestazioni. Come importante conseguenza del lavoro, presentiamo il primo dataset per il tracciamento degli stati del dialogo disponile per l'italiano, derivato da MultiWOZ 2.3.*

1 Introduction

This paper is mainly motivated by the interest of exploring, and improving, the capacity of current data-driven task-oriented conversational systems to address shifts of domain and changes of language. Our starting point is the *dialogue domain adaptation* (DDA) approach proposed by (Labruna and Magnini, 2021), which allows to adapt training dialogues collected for a source domain knowledge (e.g., restaurants in Cambridge) to a domain where certain changes (e.g., a new restaurant opens, a restaurants changes its food , etc.) have occurred. The idea behind DDA is, that, rather then trying to improve the model robustness, it is worth to generate new training dialogues that are consistent with the domain changes. In this paper we extend and experiment DDA, so that also changes of language are included, particularly moving from English to Italian.

A further motivation for our work is related to the scarcity of annotated data for task-oriented dialogues systems for the Italian language. Among the dialogic resources originally collected in Italian, we mention the recent JILDA dataset (Sucameli et al., 2020) (Sucameli et al., to appear), a corpus of dialogues in the domain of job offers, semantically annotated. NESPOLE (Mana et al., 2004) is a corpus of human-human dialogues, which is not annotated for training purposes. The LUNA dataset (Dinarelli et al., 2010), although annotated with frame information, is not publicly available. Finally, the SUGAR dataset was presented at Evalita 2018 (Di Maro et al., 2018) allowing to train systems able to interpret spoken commands in Italian for the preparation of recipes.

A second type of data is derived through automatic translation from English. This is the case of the Italian version of the SNIPS dataset (Bellomaria et al., 2019), which provides annotations for spoken language understanding and can be

Dialogue Source	Dialogue Target
USER: I am looking for a **european** food restaurant in the expensive price range. Can you help with that?	USER: Sto cercando un ristorante con cibo **osteria** nella fascia di prezzo costosa. Puoi aiutarmi?
SYS: There are **5** of those. What area do you want to dine in?	SYS: Ce ne sono **7** di questi. In quale area vorresti cenare?
USER: In the centre of town please.	USER: In centro città per favore.
SYS: How about **eraina**? Shall I book you a table?	SYS: Che ne dici di **Hosteria Il Malandrone**? Ti prenoto un tavolo?
USER: Yes, please. It will be just me and I 'd like to eat at 21:00 on the same day as my train.	USER: Sì, per favore. Sarò solo io e vorrei mangiare alle 21:00 stesso giorno del mio treno.
SYS: OK, I 've got you booked. The reference number is **VMNDMKV2** and they 'll hold you table for 15 minutes.	SYS: OK, te l'ho prenotato. Il numero di riferimento è **WPQHRNE4** ti riserveranno il tavolo per 15 minuti.

Figure 1: Example of dialogue domain and language adaptation. Words in bold indicate slot-values that have been adapted.

used for slot filling and intent detection. Similarly, an Italian version of the SQuAD question-answer pair corpus is available (Croce et al., 2019) for Italian through translation. It has to be remarked that the Italian translations of both SNIPS and SQuAD do not rely on task-oriented dialogues, rather just on question-answer pairs. In addition, they are produced through simple translation from English, i.e., preserving domain information (e.g., names of places, restaurants, hotels, etc. reflect the English domain). We also notice that, unfortunately, the recent MultiATIS++ dataset (Xu et al., 2020), does not include Italian among the languages.

We are particularly interested in data-driven approach of dialogue state tracking (DST) (Balaraman and Magnini, 2021) for the Italian language. DST captures the capacity of a model to predict the correct *dialogue state* at each turn in a dialogue, representing both the communicative goals (dialogue acts) of the user and the portion of domain knowledge involved in such goals (slot-value pairs). To the best of our knowledge, the only dataset in Italian that can be used to model dialogue state tracking is JILDA (Sucameli et al., to appear), where dialogue state annotations were carried on following the MultiWOZ style. However, being concluded very recently, still there are no available DST baselines for JILDA, and, for this reason, we have developed an Italian version of the MultiWOZ dataset (Han et al., 2020).

Starting from MultiWOZ 2.3, a popular dataset in English developed for booking traveling facilities (e.g., restaurants, hotels, trains, attractions) in the area of Cambridge, we incrementally operated both language and domain shifts. We provide three experimental configurations: (i) a translation of the Cambridge data set into Italian; (ii) a domain shift from Cambridge to Pisa, maintaining English as language; and, finally, (3) a configuration where both the initial domain and the language are changed. As a relevant side effect, the datasets for the three configurations are now available for further research on dialogue state tracking for Italian[1].

In the paper we first introduce the relavant background in *dialogue domain adaptation* (Section 2), then we explain how dialogue domain adaptation is concretely applied to domain changes, and finally we report the experiments we have conducted (Section 4 and 5).

2 Dialogue Domain Adaptation

In the *Dialogue Domain Adaptation* setting (Labruna and Magnini, 2021), we assume an initial conversational domain, represented in a KB-SOURCE, and corresponding annotated training dialogues D-SOURCE. Then, as in real application scenarios, we assume that a number of changes occur in KB-SOURCE, such that a new conversational domain KB-TARGET needs to be considered. *Dialogue domain adaptation* consists in the capacity to automatically produce new annotated dialogues D-TARGET, such that they maintain both the linguistic structure and the linguistic variability of the initial D-SOURCE dialogues, while, at the same time, being consistent with the

[1] https://github.com/tLabruna/DDA

new KB-TARGET.

Figure 1 shows an example of dialogue adaptation. On the left side we have a user-system dialogue in English grounded on the Cambridge domain, while on the right side we have the same dialogue translated into Italian and adapted to the Pisa domain. In this paper we show how to generate such adapted dialogues (i.e. D-TARGET), which differ from the original dialogues (D-SOURCE) both in language and domain. The goal is then to train a dialogue state tracking model either on D-SOURCE or D-TARGET, and to investigate the impact of such adaptations on the model performance.

2.1 Slot-Value Substitution

Following (Labruna and Magnini, 2021), we focus on domain changes due to different slot-values, while assuming the same slot-names for both the source and target domains. As for language shift, it is based on translating all the utterances in a dialogue with the exclusion of the slot-values.

Given a slot-value occurring in a source dialogue D-SOURCE, the dialogue domain adaptation process consists of choosing the best slot-value in KB-TARGET to substitute the slot-value in the D-SOURCE utterance. The first step is to check whether the slot-value is known in KB-SOURCE. If it is known, we look for a correspondence in KB-TARGET, otherwise we directly keep it in D-TARGET (or, in case of different languages, translate it into target language). In order to decide if the slot-value is in the KB-TARGET, we use a similarity function based on a variation of the Gestalt Pattern Matching algorithm (Black, 2004). We select the most similar value in the KB-TARGET and we compare it to a empirically estimated threshold. Once we found a specific slot-value in KB-SOURCE and we ensured it exceeds the threshold, the corresponding slot-value to be selected from the KB-TARGET depends on the adaptation strategy we choose to adopt.

For the experiments of this paper we have used FREQUENCY-KB, an adaptation strategy based that obtained the best performance in (Labruna and Magnini, 2021). Given a slot-value in KB-SOURCE, FREQUENCY-KB basically consists of selecting the slot-value in KB-TARGET that has the most similar frequency distribution in the KB.

3 Method

We broke down the problem of adapting a conversational dataset to a new language and a new domain into three different steps: first we performed delexicalization by inserting some placeholders in the place of the slot values; then we automatically translated the dataset, leaving the placeholders unchanged; finally, we substituted the placeholders with the new domain slot-values. Each one of these steps is discussed in the following sub-sections.

3.1 Delexicalization

The setting that we are presenting involves the annotations being specifically slot-name slot-value pairs. Both the slot-values contained in the utterances, and those in the annotations, can not be translated the same way as the rest of the text, but need to undergo a Domain Adaptation process (e.g., we don't want *I need a taxi to The Old Castle* to be translated into *Ho bisogno di un taxi per Il Vecchio Castello*).

For this reason, the first step is to *delexicalize* a D-SOURCE dialogue, i.e., substituting all the slot-values in the utterances with placeholders. The example above shows this placeholder insertion, for moving from the following original sentence:

> "I need a restaurant in the north that has Caribbean food and a moderate price range please ."

to the utterance:

> "I need a restaurant in <#0#> that has <#1#> food and a <#2#> price range please ."

3.2 Translation

The second step is to perform the translation from the source language to the target language without considering the placeholders. According to our example, we will produce the following Italian utterance:

> "Ho bisogno di un ristorante a <#0#> che abbia <#1#> cibo e un <#2#> fascia di prezzo per favore ."

3.3 Slot-Value Substitution

As a third step, the placeholders need to be substituted back with slot-values of the target domain

KB-TARGET. Which slot-values to substitute depends on the Dialogue Domain Adaptation strategy and will be discussed later.

Finally, all the slot-values - both from utterances and annotations - that could not be substituted through DDA, need to be automatically translated, which will result in the following:

> "Ho bisogno di un ristorante a est che abbia caraibico cibo e un economico fascia di prezzo per favore ."

As can be noted, a downside of using placeholders is that this method does not consider the subject-verb agreement, nor the order of the words to be different between the original and the translated text. It should also be observed that in the cases of *north* and *moderate*, the slot substitution selects different values from the KB, while in the case of *Caribbean* it could not find a correspondence in the KB, hence it got translated directly from the original.

4 Experimental Setting

We started from the public available dataset MultiWOZ 2.3 (Han et al., 2020), which consists of a collection of more than ten thousand annotated dialogues (with dialogue states) spanning over seven domains related to traveling in Cambridge (e.g., restaurants, hotels, attractions, trains).

Pisa KB-TARGET. We manually created a KB-TARGET for Pisa, mirroring the instance distribution of the KB-SOURCE for Cambridge. For every entity instance of the Cambridge KB, a corresponding Pisa instance was created, keeping the slot-names as they were in the original, and changing only the slot-values. The specific instances were chosen by analysing the frequency distribution in the Cambridge KB and finding a similar correlation in the Pisa domain. For example, all the Cambridge restaurants with INDIAN food type, which is the most common in Cambridge, were substituted with Pisa restaurants with ITALIAN food type, which is the most common in Pisa. All the Pisa instances were taken from publicly available datasets containing real information on Pisa entities [2].

Automatic translation. As for translation from English to Italian, we used the automatic translator available at FBK. [3] The MT engine is built on the ModernMT framework[4] which features neural machine translation implementing the Transformer architecture (Vaswani et al., 2017). A big model (more than 200 million parameters) is trained on generic domain data, taken from the OPUS repository[5].

Test data used in the experiments were manually checked, correcting a number of translation issues, including, for instance, wrong prepositions used for time expressions (from *di 13:00* to *delle 13:00*), and wrong agreements (from *prezzi medio* to *prezzi medi*). Training data were not corrected.

Datasets. We run experiments over the following four datasets:

- CAM-ENG. This is the original MultiWOZ 2.3 dataset, with Cambridge as domain and English as language. It is used as referent for the other experiments.
- CAM-ITA. This is the translation to Italian of the original MultiWOZ 2.3 dataset, with Cambridge as domain.
- PISA-ENG. This is the original MultiWOZ 2.3 dataset adapted to the new Pisa knowledge base, using dialogue domain adaptation, as described in Section 3.
- PISA-ITA. This is the MultiWOZ 2.3 dataset, first translated into Italian and then domain adapted to the Pisa knowledge base.

For all the datasets we kept the same training/test split of dialogues as in the original MultiWOZ 2.3. In addition, we have experimented the following combinations:

- CAM-ITA + CAM-ENG. This combination provides all the available data for the Cambridge domain, mixing the two languages.
- PISA-ENG + CAM-ENG. This combination provides all the available data for English, mixing the two domains.
- CAM-ITA + PISA-ITA. This combination provides all the available data for Italian, mixing the two domains.

[2] http://www.datiopen.it/

[3] We would like to thank the Machine Translation Research Unit of FBK, and in particular Mauro Cettolo, for the kind support in the generation of automatic translations.

[4] http://github.com/modernmt/modernmt

[5] http://opus.nlpl.eu

Training	Test	Training Accuracy	Turn Accuracy	Joint F1	Joint Accuracy
Cam-ENG	Cam-ENG	0.52	0.97	0.9	0.49
Cam-ITA + Cam-ENG	Cam-ENG	0.48	0.97	0.9	0.49
Pisa-ENG + Cam-ENG	Cam-ENG	0.54	0.97	0.9	0.49
Cam-ITA	Cam-ITA	0.42	0.95	0.87	0.4
Cam-ITA + Cam-ENG	Cam-ITA	0.48	0.96	0.88	0.42
Cam-ITA + Pisa-ITA	Cam-ITA	0.4	0.95	0.87	0.38
Pisa-ENG	Pisa-ENG	0.54	0.97	0.89	0.5
Pisa-ITA + Pisa-ENG	Pisa-ENG	0.49	0.97	0.91	0.52
Pisa-ENG + Cam-ENG	Pisa-ENG	0.54	0.97	0.91	0.52
Pisa-ITA	Pisa-ITA	0.39	0.95	0.86	0.37
Pisa-ITA + Pisa-ENG	Pisa-ITA	0.49	0.96	0.88	0.42
Cam-ITA + Pisa-ITA	Pisa-ITA	0.4	0.95	0.86	0.37

Table 1: Performance of the TRADE algorithm over the datasets used in the experiments.

- PISA-ITA + PISA-ENG. This combination provides all the available data for the Pisa domain, mixing the two languages.

Dialogue State Tracking Model. The goal of the experiments is to assess the robustness of a dialogue state tracking model when domain and language are changed. As for DST model, we have used TRADE (Wu et al., 2019), an algorithm optimized for being used on multi-domain dialogues such MultiWOZ.

5 Results

Results of the experiments are presented in Table 1. The first column indicates which dataset the model was trained on; the second column reports the dataset used for testing the model; the last four columns report measures on the model performance. Training Accuracy refers to the Joint Accuracy obtained at training time; Turn Accuracy indicates how many single predictions were actually correct; the Joint F1 score reflects the accuracy of the model, considering both precision and recall; finally, the Joint Accuracy, measures the percentage of correct predictions of dialogue states for every dialogue turn, where a prediction is considered correct if all the slot values in the dialogue turn are correctly predicted. Results are reported into four groups depending on the dataset that has been used for testing. For every group we have three configurations: the first experiment reports the performance with the initial dataset, the second considers the extension of the initial dataset with the second language, and finally, the third experiment considers the extension of the initial dataset with the second domain.

6 Discussion

Results reveal several interesting aspects. First, we register a decrease in performance between the datasets in English and those automatically translated to Italian. This can be due to the process of placeholder insertion and subsequent substitution of slot-values, along with the translation itself, which can be source of errors. On the other side, the domain adaptation from CAM-ENG to PISA-ENG and from CAM-ITA to PISA-ITA did not show the same decrease of performance, rather it resulted even in a small increase for the first case.

The central part of our work, however, focused on generating adapted dialogues and investigating the performance variations derived from them.

Slot-name	Cam-ITA Accuracy	Cam-ITA vs Pisa-ITA Overlap	Cam-ITA + Pisa-ITA Accuracy	Cam-ITA vs Cam-ENG Overlap	Cam-ITA + Cam-ENG Accuracy
Train-departure	0.925	0.421	0.924	0.607	0.934
Train-destination	0.950	0.466	0.947	0.762	0.956
Restaurant-area	0.846	0.561	0.892	0.051	0.851
Hotel-area	0.787	0.812	0.811	0.03	0.795

Table 2: Slot-name accuracy prediction with comparison to the overlap of the slot-name between the dialogues. The first column is the considered slot-name. The second column is the predicted accuracy of the slot given by the TRADE model trained on Cam-ITA and tested on Cam-ITA. The third and fourth columns show the overlap and the prediction accuracy with respect to domain change. The remaining columns show the same measures for the language change.

With regards to this aspect, it should be noted that the addition of a second language resulted in a significant improvement almost in all cases, with an increase of 5% for CAM-ITA, 4% for PISA-ENG and 13.5% for CAM-ITA. On the other side, the addition of the second domain does not bring much advantage, resulting in zero change for CAM-ENG and PISA-ITA, a small decrease for CAM-ITA and a small increase for PISA-ENG.

6.1 Overlaps Between Datasets

In order to better understand the factors that affect the variation of Joint Accuracy performances between the datasets of each group, we have analysed the overlaps among the training datasets. We estimated such overlap as the proportion of slot-values in two datasets for every domain that are exactly the same .

We have observed that in most of the cases adding a dataset with high overlap for a certain domain produces an improvement in DST performance for that domain. As an example, the domain with highest overlap between the Cam-ITA dataset and the Pisa-ITA dataset is *Taxi* (86.11% of overlap). On the other side, the domain with lowest overlap between the same datasets is *Attraction* (44.45% of overlap). These overlaps have strong correlation with the DST performances on the two domains: the Cam-ITA + Cam-ENG dataset produces an improvement of 1.5 points with respect to the Cam-ITA dataset on the *Taxi* domain, and shows a decrease of 1 point on the *Attraction* domain.

This correlation can also be verified if we look at a slot-name level. Table 2 shows some examples of slot-names with corresponding overlaps between dialogues and slot-name prediction accuracy, taken from the Cam-ITA setting with domain and language additions. As it can be noted, when the slot-name overlap between the aggregated dialogue and Cam-ITA is higher, the respective prediction accuracy also tends to be higher.

7 Conclusion

We have investigated domain and language shift for data-driven task-oriented dialogue systems. We have extended recent work on *dialogue domain adaptation* to a cross-language setting, where both the domain and the language are changed. We showed that: (i) state-of-the-art models are very sensible to language shift obtained through automatic translation; (ii) combining training data for the two languages is always beneficial; on the contrary, combining data of different domains does not produce any improvement in all of our settings. Finally, as a relevant side effect of our work, we present a new dataset for dialogue state tracking available for Italian, derived from MultiWOZ 2.3. All the data are made available for further research on dialogue domain adaptation.

References

V. Balaraman and B. Magnini. 2021. Domain-aware dialogue state tracker for multi-domain dialogue systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:866–873.

Valentina Bellomaria, Giuseppe Castellucci, Andrea Favalli, and Raniero Romagnoli. 2019. Almawave-slu: A new dataset for SLU in italian. In Raffaella Bernardi, Roberto Navigli, and Giovanni Semeraro, editors, *Proceedings of the Sixth Italian Conference on Computational Linguistics, Bari, Italy, November 13-15, 2019*, volume 2481 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Paul E Black. 2004. Ratcliff/obershelp pattern recognition. *Dictionary of algorithms and data structures*, 17.

Danilo Croce, Alexandra Zelenanska, and Roberto Basili. 2019. Enabling deep learning for large scale question answering in italian. *Intelligenza Artificiale*, 13(1):49–61.

Maria Di Maro, Antonio Origlia, and Francesco Cutugno, 2018. *Overview of the EVALITA 2018 Spoken Utterances Guiding Chef's Assistant Robots (SUGAR) Task*, pages 79–85. 01.

Marco Dinarelli, Evgeny Stepanov, S. Varges, and Giuseppe Riccardi. 2010. The luna spoken dialogue system: Beyond utterance classification. pages 5366 – 5369, 04.

Ting Han, Ximing Liu, Ryuichi Takanobu, Yixin Lian, Chongxuan Huang, Wei Peng, and Minlie Huang. 2020. Multiwoz 2.3: A multi-domain task-oriented dataset enhanced with annotation corrections and co-reference annotation. *arXiv preprint arXiv:2010.05594*.

Tiziano Labruna and Bernardo Magnini. 2021. Addressing slot-value changes in task-oriented dialogue systems through dialogue domain adaptation. In *Proceedings of RANLP 2021*.

Nadia Mana, Roldano Cattoni, Emanuele Pianta, Franca Rossi, Fabio Pianesi, and Susanne Burger. 2004. The Italian NESPOLE! corpus: a multilingual database with interlingua annotation in tourism and medical domains. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, May. European Language Resources Association (ELRA).

Irene Sucameli, Alessandro Lenci, Bernardo Magnini, Maria Simi, and Manuela Speranza. 2020. Becoming JILDA. In Johanna Monti, Felice Dell'Orletta, and Fabio Tamburini, editors, *Proceedings of the Seventh Italian Conference on Computational Linguistics, CLiC-it 2020, Bologna, Italy, March 1-3, 2021*, volume 2769 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Irene Sucameli, Alessandro Lenci, Bernardo Magnini, Manuela Speranza, and Maria Simi. to appear. Toward data-driven collaborative dialogue systems: The jilda dataset. *Italian Journal of Computational Linguistics*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. 2019. Transferable multi-domain state generator for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 808–819, Florence, Italy, July. Association for Computational Linguistics.

Weijia Xu, Batool Haider, and Saab Mansour. 2020. End-to-end slot alignment and recognition for cross-lingual NLU. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5052–5063, Online, November. Association for Computational Linguistics.

Agentività e telicità in GilBERTo: implicazioni cognitive

Agnese Lombardi, Alessandro Lenci
Università di Pisa, Pisa, Italia
a.lombardi19@studenti.unipi.it, alessandro.lenci@unipi.it

Abstract

English. The goal of this study is to investigate whether a Transformer-based neural language model infers lexical semantics and use this information for the completion of morphosyntactic patterns. The semantic properties considered are telicity (also combined with definiteness) and agentivity. Both act at the interface between semantics and morphosyntax: they are semantically determined and syntactically encoded. The tasks were submitted to both the computational model and a group of Italian native speakers. The comparison between the two groups of data allows us to investigate to what extent neural language models capture significant aspects of human semantic competence.

Italiano. *L'obiettivo di questo studio è quello di indagare se neural language models basati su Transformer inferiscono aspetti semantico-lessicali rilevanti per l'interfaccia con la sintassi ed utilizzano queste informazioni per il completamento di task morfosintattici. Le proprietà semantiche considerate sono la telicità (anche in relazione all'individuazione) e l'agentività. Entrambe sono semanticamente determinate e sintatticamente codificate. I task sono stati sottoposti sia al modello che ai parlanti. La comparazione tra i due gruppi di dati ci permetterà di determinare se questi modelli computazionali catturano aspetti significativi della competenza semantica umana.*

1 Introduzione

L'ipotesi distribuzionale stabilisce che lessemi con contesti linguistici simili hanno un significato simile (Wittgenstein, 1953; Harris, 1954; Firth, 1957).

I modelli distribuzionali sono stati impiegati con successo in molti task di Natural Language Processing, ma quale siano le conoscenze acquisite durante il processo di addestramento rimane una questione ancora aperta.

Uno degli approcci per comprendere la natura di queste informazioni linguistiche consiste nel valutare la loro accuratezza in task psicolinguistici[1]. Alcuni studi hanno indagato proprietà e dipendenze sintattiche (Linzen et al., 2016; Ettinger, 2016; Wilcox et al., 2018; Futrell et al., 2019; Marvin e Linzen, 2018; Hu et al., 2020; Lau et al., 2020), altri si sono concentrati su aspetti semantici e pragmatici come: la similarità (Hill et al.,2015), la categorizzazione (Baroni e Lenci, 2010), l'analogia (Mikolov et al., 2013), la negazione (Marvin e Linzen, 2018; Jumelet e Hupkes, 2018), il ragionamento pragmatico, i ruoli semantici e la conoscenza eventiva (Ettinger, 2020).

Il nostro lavoro contribuisce a questa linea di ricerca e ne adotta l'approccio psicolinguistico, ma se ne discosta nelle proprietà investigate, proponendo l'analisi della telicità (in combinazione con l'individuazione) e dell'agentività.

L'obiettivo è indagare se l'inferenza di queste proprietà semantico-lessicali favorisce l'elaborazione di alcuni task morfosintattici. Nella nostra analisi abbiamo scelto di utilizzare un modello distribuzionale di tipo predittivo che utilizza rappresentazioni contestualizzate (Peters et al., 2018; Devlin et al., 2019): GilBERTo, un modello distribuzionale italiano ispirato all'architettura di RoBERTa (Liu et al., 2019).

2 Aspetti della semantica di interfaccia: azionalità e agentività

Un importante dominio dell'informazione lessicale riguarda l'evento e i suoi partecipanti. Se da un lato l'aspetto è una nozione di natura eminentemente morfologica e semantica, che riguarda le

[1] Gli stimoli, nei task psicolinguistici, sono progettati in modo da fornire informazioni sulle proprietà linguistiche che influenzano il comportamento umano (giudizio grammaticale, velocità di letture o risposte neurali).

modalità di svolgimento dell'evento (piuttosto che la sua localizzazione e la serie di rapporti temporali); l'*azionalità verbale*, d'altra parte, non viene codificata dalla morfologia flessiva. Non basta dire che l'azionalità sia un fatto inerente al significato intrinseco di un lessema, bisogna individuare delle classi coerenti di verbi, contraddistinte da un comportamento sintattico omogeneo nell'ambito della lingua considerata. Ci sono diversi aspetti lessicali che possono essere codificali da una classe verbale e che pertengono alla classificazione azionale.

Vendler (1967) categorizza le classi azionali sulla base di tre proprietà fondamentali (la duratività, la dinamicità e la telicità) e individua quattro gruppi principali: i verbi stativi (*states*), di attività (*activities*), risultativi (*accomplishments)* e trasformativi (*achievements)*. I verbi trasformativi e quelli risultativi vengono raggruppati nella categoria dei *telici*. Gli eventi telici hanno la caratteristica di essere finalizzati al raggiungimento di un telos, ovvero una meta o una fine.

2.1 Telicità

Finora abbiamo considerato la classe dei telici come l'insieme dei risultativi e dei trasformativi. In ogni caso, è necessario specificare che la telicità può essere intesa come un continuum, un asse semantico che vede ai due estremi i prototipi della categoria (inerentemente telici e inerentemente atelici) e al centro gli elementi che definiremo *configurazionali*[2]. Dunque, la telicità non è una proprietà discreta e non sempre è possibile definirla in maniera inequivocabile (non essendo determinata solo dai tratti lessicali): essendo fortemente dipendente dal contesto (dagli argomenti del verbo e dalla transitività, ma anche dalla coniugazione dell'aspetto verbale) e veicolata dal senso complessivo della frase. Ad esempio, "disegnare" e "cantare" sono di per sé predicati verbali non telici: ciò che li rende telici, in un determinato contesto, è la presenza di un oggetto diretto che li determina, finalizzandoli al raggiungimento di un preciso scopo.

In "Gennaro ha disegnato/ha cantato tutto il pomeriggio" il predicato verbale si configura come atelico, ma diventa telico se si ha "Gennaro ha disegnato il ritratto di mia nonna" o "Gennaro ha cantato la sua canzone preferita".

Da Bertinetto (1997), apprendiamo che un test per distinguere l'accezione telica di un verbo da quella non-telica è l'aggiunta dell'avverbiale "in x tempo"[3] che risulta incompatibile con i predicati non telici. L'applicazione dell'avverbiale "per X tempo", invece, risulta o incompatibile con i predicati verbali telici o se compatibile, ne neutralizza la telicità.

2.2 Telicità e individuazione

L'individuazione dell'oggetto o del soggetto può incidere sull'interpretazione telica assegnata all'evento. Il concetto di individuazione unifica diverse proprietà dell'argomento e può essere considerata anch'essa una proprietà semantica di interfaccia, perché agisce sia a livello semantico che morfosintattico. L'individuazione si riferisce alla propensione di un'entità ad essere concepita come un individuo indipendente. Possiamo considerare l'individuazione come la risultante delle seguenti proprietà: individui, animatezza, concretezza/astrattezza, singolare/plurale, mass/count, referenziale/non-referenziale (Romagno, 2005). Concependo l'individuazione come un continuum, i significati possono essere raggruppati secondo classi di equivalenza che condividono le stesse proprietà di individuazione e le classi di individuazione possono essere ordinate sulla base del loro grado di individuazione. Il grado di individuazione di un'entità può essere calcolato sulla base della media derivata tramite l'unione dei valori di tutti i fattori che la determinano. Considereremo [+ individuato] un argomento umano, proprio, animato, concreto, singolare, numerabile, referenziale e [- individuato] un argomento inanimato, comune, astratto, plurale, non numerabile, non referenziale. Sia dal punto di vista semantico, sia da quello morfosintattico, si registra un'influenza reciproca tra individuazione e telicità. Ad esempio, nella frase "mangiare del pane", l'oggetto è poco individuato; nella frase "mangiare una pagnotta di pane", invece, l'oggetto è individuato e rappresenta l'argomento interno diretto del predicato. Ne consegue che nel primo caso l'interpretazione assegnata all'evento è atelica e nel secondo caso è telica.

2.3 Agentività

Secondo Cruse (1971) l'agentività è presente in ogni frase che si riferisce ad un'azione effettuata

[2] Con questo termine faremo riferimento, d'ora in poi, a quei predicati verbali la cui interpretazione telica (o atelica) è determinata dal contesto (in particolare dall'individuazione dell'oggetto o del soggetto).

[3] "X tempo" simboleggia un'espressione temporale numericamente quantificata: in due minuti, in due giorni, in due ore, in due anni…

da un'entità che impiega la propria energia per condurre l'azione. Nella definizione di entità sono inclusi gli esseri viventi, alcuni tipi di macchine ed eventi naturali. Da ciò è possibile dedurre che l'argomento agentivo è prototipicamente il soggetto, essendo esso il promotore dell'azione, ed è sempre associato con una struttura logica[4] di attività; e che solo verbi che possiedono nella loro struttura logica un predicato di attività possono avere un argomento agentivo. Nella struttura logica di un predicato, l'agentività è rappresentata come DO (x, [do (x, ...]. Ad esempio, se si confrontano i verbi "kill" e "murder" (il primo verbo può accogliere soggetti inanimanti, mentre il secondo no) la struttura logica si configura come: kill: [do (x, Ø)] CAUSE [BECOME dead (y)] murder: DO (x, [do (x, Ø] CAUSE [BECOME dead (y)]) (Pustejovsky e Batiukova, 2019).

Ci sono altri verbi, però, che possono assumere un'interpretazione agentiva. Infatti, il più delle volte, l'agentività è determinata dal modo in cui un verbo è utilizzato all'interno di una frase e non è un'inerente proprietà lessicale del verbo. In questi casi, l'agentività non fa parte del significato lessicale del verbo e non è rappresentata nella sua struttura logica, piuttosto è determinata da implicazioni basate sull'animatezza dell'attore e sulle proprietà lessicali del verbo. Holisky (1987) sostiene che l'interpretazione agentiva spesso sorge dall'intersezione tra le proprietà semantiche all'interno di una frase (le proprietà semantiche dell'attore NP e del predicato) e i principi generali di conversazione.

Un test molto semplice per capire se l'agentività è lessicalizzata in un verbo coinvolge l'avverbio "inavvertitamente" e consiste nel verificare se il suo impiego crea una contraddizione all'interno della frase. Se la frase diventa contraddittoria, allora il predicato verbale lessicalizza l'agentività.

È il caso di: "Gennaro ha assassinato *inavvertitamente il suo vicino", in cui la contraddizione è evidente, quindi il predicato è agentivo. Anche l'agentività, come la telicità, è una proprietà che agisce nell'interfaccia tra sintassi e semantica.

3 Esperimento

Il nostro obiettivo è, quindi, indagare se GilBERTo è in grado di inferire la telicità (anche in combinazione con l'individuazione) e l'agentività e di utilizzare questa inferenza per il completamento di task morfosintattici. Inoltre, vogliamo determinare se l'elaborazione del modello può essere comparata a quella dei parlanti nei medesimi task.

Essendo entrambe queste proprietà semantiche, codificate morfosintatticamente, possiamo determinarne la corretta elaborazione mediante dei test morfosintattici. La risposta selezionata sarà dunque informativa dal punto di vista semantico.

Per garantire un confronto diretto tra il modello e i parlanti, ad entrambi verranno sottoposti i medesimi task.

3.1 Stimoli

I soggetti e il modello dovevano completare dei cloze test con la giusta opzione morfosintattica.

Abbiamo ideato tre task. Il primo task indaga la telicità, il secondo l'individuazione in rapporto alla telicità e il terzo l'agentività. Ogni task è composto da sessanta frasi affermative con verbo coniugato al passato prossimo.

Nel primo task sulla telicità le frasi dovevano essere completate con la preposizione "in" o "per" nelle locuzioni avverbiali "in/per X tempo". I soggetti sono nomi comuni, impiegati alla terza persona, animati e, a volte, utilizzati con il supporto di un aggettivo possessivo. Abbiamo incluso verbi inerentemente telici (sia risultativi che trasformativi), verbi inerentemente atelici e verbi configurazionali (20 + 20 + 20). Nelle seguenti frasi, estratte dal primo task, riportiamo esempi con verbo telico (1), atelico (2) e configurazionale (3):

(1) L'operaio ha demolito la casa in/per un'ora
(2) Mia sorella ha dormito in/per tre ore
(3) Il ragazzo ha corso in/per un'ora

Nel secondo task, che indaga la telicità in relazione all'individuazione, abbiamo utilizzato lo stesso cloze test. Tuttavia, abbiamo strutturato un design fattoriale che divide le frasi in quattro gruppi (di 15 frasi), secondo lo schema seguente:

I gruppo: soggetto [+ ind][5] e oggetto [- ind]
II gruppo: soggetto [+ ind] e oggetto [+ ind]
III gruppo: soggetto [- ind] e oggetto [- ind]
IV gruppo: soggetto [- ind] e oggetto [+ ind][6]

[4] "[…] Logical Structures (LS) consisting of constants, which mostly represent predicates, and modifiers (BECOME, INGR, CAUSE, etc.). […] these elements are not words from any natural language, but items of a semantic metalanguage" (Van Valin e LaPolla, 1997).

[5] Ind = individuato

[6] Per i soggetti [+ individuati] abbiamo utilizzato nomi comuni di persona con aggettivi possessivi; per i soggetti [- individuati] nomi comuni plurali o nomi astratti. Gli oggetti [- individuati] sono costituiti da nomi comuni (riferiti a liquidi, plurali o nomi massa) con un aggettivo qualificativo senza articolo determinativo

In ogni gruppo abbiamo incluso predicati verbali telici, atelici e configurazionali (5+5+5). Riportiamo una frase per ognuno dei quattro gruppi:

I Mio fratello ha bevuto latte fresco in/per cinque minuti

II Mio fratello ha bevuto un bicchiere di latte in/per cinque minuti

III I mobili hanno accumulato della polvere densa in/per dieci anni

IV I mobili hanno accumulato un sacco di polvere in/per dieci anni

Nel terzo task, che indaga l'agentività, le frasi dovevano essere completate con "inavvertitamente" o "intenzionalmente". Abbiamo variato sia le proprietà del soggetto (includendo soggetti con il ruolo prototipico di actor, ma anche soggetti meno prototipici) e quelle dell'oggetto (includendo oggetti con il ruolo prototipico di undergoer, ma anche oggetti meno prototipici). Abbiamo incluso predicati verbali che hanno la proprietà dell'agentività lessicalizzata nella loro struttura semantica (quindi inerentemente agentivi), predicati verbali inerentemente inagentivi e predicati verbali che possono assumere entrambi i valori a seconda del contesto (20 + 20 + 20). Anche in questo caso abbiamo escluso i nomi propri ed i soggetti sono tutti animati ed alla terza persona. Il seguente esempio riporta rispettivamente un verbo agentivo, inagentivo e configurazionale:

(4) Mio fratello ha deciso intenzionalmente/inavvertitamente di scegliere

(5) Mio fratello è invecchiato intenzionalmente/inavvertitamente

(6) Mio padre ha cotto intenzionalmente/inavvertitamente per molto tempo la carne

Nei primi due task, le frasi sottoposte al modello, contenevano una parola mascherata [7] nell'input e il modello doveva fornire come output, al suo posto, le prime cinque opzioni più probabili e le relative probabilità. Nel terzo task invece, il modello fornisce come output direttamente la frase completa con una delle opzioni. I parlanti, invece, dovevano scegliere in ognuno dei tre task l'opzione preferibile tra le due proposte.

3.2 Partecipanti

65 volontari madrelingua italiani avevano il compito di completare le frasi scegliendo l'opzione più opportuna. Ai parlanti venivano fornite le istruzioni per il completamento dei task all'inizio degli stessi. Tutti i dati sono stati raccolti tramite Google Forms.

3.3 Modello

GilBERTo è un modello del linguaggio italiano preaddestrato basato sull'architettura di RoBERTa e sull'approccio di tokenizzazione del testo di CamemBERT. Il modello è stato addestrato con la tecnica di mascheramento delle subwords per 100k passi gestendo 71 GB di testo italiano con 11.250.012.896 parole (OSCAR: Open Super-large Crawled Almanach coRpus). È stato considerato un vocabolario di 32k BPE (Byte-Pair Encoding) subwords, generate usando SentencePiece tokenizer. Nei primi due task è stata utilizzata la libreria pytorch\fairseq Python e nel terzo task la libreria FitBERT.

4 Risultati

Le teorie linguistiche stabiliscono che i verbi inerentemente telici dovrebbero selezionare "in x tempo", mentre gli inerentemente atelici "per x tempo". I verbi configurazionali selezionano "in" o "per" a seconda dell'interpretazione telica che il soggetto vuole conferire alla frase. I dati del primo task confermano questo schema, come illustra la tabella 1, in cui sono riportate le preferenze del modello[8] e dei parlanti.

Predicati	**Modello (%)**		**Parlanti (%)**	
	in	**per**	**in**	**per**
Telici	70	20	100	0
Atelici	80	10	0	100
Config.	35	30	50	50

Table 1: Primo task

Come si evince, i dati dei verbi inerentemente telici e inerentemente atelici di entrambi i gruppi presentano pochissima dispersione: l'interpretazione è telica o atelica a seconda del predicato verbale e non c'è indecisione tra le opzioni proposte.

("latte fresco"); con gli oggetti [+ individuati] si assiste allo schema opposto: nomi al singolare, con articolo determinativo, o nomi leggeri quantificatori ("un sacco di polvere"). Per favorire una comparazione diretta tra due frasi abbiamo utilizzato lo stesso predicato verbale e lo stesso soggetto (nei primi due gruppi [+ individuato] e negli ultimi due [- individuato]) e abbiamo variato solo l'individuazione dell'oggetto.

[7] Ad esempio: *Il ragazzo ha corso <mask> un'ora.*

[8] Le percentuali del modello, nelle tabelle 1 e 2, corrispondono al valore della mediana, calcolata in relazione alle probabilità fornite dal modello come output.

I verbi configurazionali invece, presentano, sia nel modello che nei parlanti, una dispersione dei dati più ampia e nessuna delle due opzioni risulta preferibile.

I dati del secondo task, raccolti nella tabella 2, presentano uno scenario più complesso.

Predicati	Modello (%)		Parlanti (%)	
	in	**per**	**in**	**per**
Telici (I gruppo)	60	15	45	55
Telici (II gruppo)	35	10	100	0
Telici (III gruppo)	60	15	80	20
Telici (IV gruppo)	30	20	100	0
Atelici (I gruppo)	0	80	0	100
Atelici (II gruppo)	20	80	80	20
Atelici (III gruppo)	10	60	40	60
Atelici (IV gruppo)	15	55	75	25
Config. (I gruppo)	0	90	0	100
Config. (II gruppo)	10	70	100	0
Config. (III gruppo)	20	40	30	70
Config. (IV gruppo)	20	60	40	60

Table 2: Secondo task

I dati mostrano che il modello seleziona "in" solo con i verbi inerentemente telici (si registra uno scarto maggiore tra le due opzioni nel primo e nel terzo gruppo, in cui l'oggetto è [- individuato]). I parlanti, invece, con verbi inerentemente telici selezionano "in" in ognuno dei gruppi, tranne che nel primo (soggetto [+ individuato] e oggetto [- individuato]), in cui "per" viene preferito nel 55% dei casi. Contrariamente al modello, in cui, nel secondo e nel quarto gruppo (con oggetti [+ individuati]), "in" ottiene una probabilità vicina a quella di "per", nei parlanti è del 100% ("per" non viene mai selezionato).

Con verbi inerentemente atelici, invece, i parlanti selezionano "in" quando l'oggetto è [+ individuato] e "per" quando l'oggetto è [- individuato] (nel secondo e nel quarto gruppo, quindi, rispettivamente nell'80% e nel 75% dei casi). Nel modello, invece, l'opzione "per" risulta preferibile in ognuno dei quattro casi considerati. Infine, con verbi configurazionali i parlanti mostrano una preferenza per "in" nel secondo gruppo (soggetto e oggetto [+ individuati]) e per "per" nei restanti tre. Nello specifico, però, nel primo gruppo (soggetto [+ individuato] e oggetto [- individuato]) "per" risulta vincente nel 100% dei casi (confermando i dati dei verbi inerentemente telici, in cui, nel primo gruppo, i parlanti selezionavano "per" al 55%); mentre, nel terzo e nel quarto gruppo (con soggetti [- individuati]), "per" ottiene percentuali più basse, determinando conseguentemente uno scarto inferiore tra le due opzioni. Il modello rispetta lo schema dei parlanti con la variazione delle probabilità di "per" tra primo gruppo (con soggetto [+ individuato] ha un valore del 90%) e terzo e quarto (con soggetto [-individuato] ha rispettivamente il 40% e il 60%); ma non conferma lo schema dei parlanti nel secondo gruppo (ha il 70%, mentre dai parlanti non veniva mai scelto).

Nella tabella 3 sono raccolti i dati[9] del terzo task.

Predicati	Modello (%)	Parlanti (%)
Agentivi	inavv. (70)	intenz. (100)
Inagentivi	inavv. (65)	inavv. (100)
Config.	inavv. (60)	inavv. (50)

Table 3: Terzo task

I risultati del terzo task mostrano che il modello sceglie l'opzione "inavvertitamente" in più dl 50% delle frasi, per ognuno dei tre gruppi di verbi, nonostante la variazione di agentività. I parlanti, invece, mostrano coerenza con le ipotesi linguistiche.

5 Discussione

L'analisi aveva lo scopo di testare l'elaborazione della telicità (anche in rapporto all'individuazione) e dell'agentività, sia nei parlanti che nel modello, e di indagare se quest'elaborazione determina il giusto completamento morfosintattico.
I dati mostrano che i parlanti operano in maniera coerente con l'ipotesi proposta e con la teoria linguistica. Infatti, è la telicità a determinare la giusta codifica morfosintattica. Inoltre, mostrano un'influenza evidente dell'individuazione sull'interpretazione telica. In presenza di verbi inerentemente telici i parlanti selezionano "in" senza essere influenzati dalla minore individuazione del soggetto. L'unico caso in cui la valenza telica inerente del predicato subisce una variazione è con soggetto [+ individuato] e oggetto [- individuato]. Questo stesso schema non si riscontra con soggetto [- individuato] e oggetto [- individuato].

Sappiamo che l'oggetto riveste prototipicamente il ruolo di paziente e che quindi è colui che nel caso di un evento telico subisce il mutamento

[9] Le percentuali del modello, nella tabella 3, corrispondono al numero di frasi in cui il modello ha preferito inavvertitamente o intenzionalmente.

di stato, quindi [+ coinvolto] e [+ individuato][10]. D'altra parte, il soggetto è prototipicamente il promotore dell'azione, quindi [- coinvolto] e [- individuato] dell'oggetto. In questo caso, possiamo supporre che a guidare l'interpretazione atelica (nonostante la telicità inerente del predicato) sia la non prototipicità dei due argomenti nella frase[11]. A riprova, ciò non si verifica nelle frasi del terzo gruppo, in cui non vi è differenza tra coinvolgimento ed individuazione del soggetto e dell'oggetto.

Anche i dati comportamentali dei verbi inerentemente atelici risultano coerenti con l'ipotesi: i parlanti conferiscono un'interpretazione telica alle frasi in cui l'oggetto è [+ individuato] e atelica a quelle in cui è [- individuato]. Con i verbi configurazionali l'interpretazione telica è possibile solo se entrambi gli argomenti sono individuati. I dati comportamentali, inoltre, riflettono la natura scalare della telicità: i verbi configurazionali sono quelli che riportano una dispersione dei dati più ampia.

Questo tipo di elaborazione si riscontra anche nel modello, dove i verbi configurazionali non mostrano nessuna preferenza netta a favore di una delle due opzioni, dimostrando che il modello riesce ad inferire la natura scalare della telicità. Tuttavia, una differenza emerge nell'elaborazione dell'individuazione. Da un lato i parlanti sono influenzati dall'individuazione nell'interpretazione telica; d'altra parte, il modello non mostra la stessa sensibilità. Questa mancata elaborazione dell'individuazione viene confermata dal fatto che con verbi inerentemente telici il modello predilige sempre un'interpretazione telica e con verbi inerentemente atelici un'interpretazione atelica. Quindi l'accordo tra modello e parlanti è determinato dalle proprietà del predicato e non dall'individuazione di soggetto e oggetto.

Anche per l'agentività i parlanti mostrano coerenza con l'ipotesi e con le teorie linguistiche. Nei casi in cui l'agentività è codificata nell'informazione eventiva del verbo viene selezionato con una preferenza netta l'avverbio ad essa associato. Viceversa, accade con verbi inerentemente inagentivi; mentre i verbi configurazionali non mostrano preferenza per nessuna delle due opzioni. Il modello, al contrario, completa il task senza essere influenzato dall'agentività. Questo risultato potrebbe essere determinato dal tipo di task o dall'utilizzo di FitBERT, che per la prima volta viene applicato ad un modello basato sull'architettura di RoBERTa.

Generalizzando, il modello riesce ad utilizzare le proprietà semantiche veicolate dal predicato per determinare la giusta codifica morfosintattica: elabora, quindi, la telicità in coerenza con le teorie linguistiche, come una proprietà scalare. Tuttavia, non si può affermare che questa elaborazione avvenga anche per le proprietà semantiche che sono veicolate dal contesto dell'intera frase: l'agentività o la variazione della telicità dovuta all'individuazione.

6 Conclusioni

La differenza di elaborazione tra modello e parlanti ci permette di proporre delle implicazioni dal punto di vista teorico. La prima implicazione è che seppure questi modelli mostrino una certa sensibilità e una certa aderenza al modo in cui i parlanti processano il linguaggio, non possono essere considerati un modello cognitivo di elaborazione del linguaggio. Tuttavia, questa analisi ci permette di ipotizzare la codifica di queste proprietà di semantica lessicale nell'informazione vettoriale dei modelli distribuzionali, facendo luce su quali sono le informazioni semantiche codificate.

Sicuramente esiste un'influenza distribuzionale nel modo in cui i parlanti utilizzano le informazioni, ma bisogna considerare anche fattori che dipendono dal contesto extralinguistico. In lavori futuri ha senso continuare ad indagare l'elaborazione di proprietà di semantica lessicale nei modelli distribuzionali, magari adottando altre tecniche di indagine e confrontando dati estratti da modelli diversi. Futuri lavori potranno indagare altre proprietà di semantica lessicale: ad esempio l'intransitività scissa.

Inoltre, il nostro lavoro può essere migliorato includendo lo studio dell'aspetto verbale e dell'influenza che questo ha nell'interpretazione della frase (coniugando le frasi non solo all'aspetto perfettivo, ma anche all'imperfettivo). Ad esempio, potrebbe essere interessante considerare il caso delle lingue slave che grammaticalizzano la telicità attraverso l'opposizione tra aspetto perfettivo e imperfettivo. Infine, questi studi potrebbero essere utilizzati per implementare questi modelli distribuzionali, migliorando il modo in cui veicolano la composizionalità semantica (a livello frasale).

[10] Telicità, coinvolgimento e individuazione dell'oggetto sono anche alcuni dei parametri che determinano la transitività di una frase.

[11] Il soggetto, in questo caso, ha le proprietà semantiche prototipiche dell'oggetto e, quest'ultimo, ha quelle prototipiche del soggetto.

References

Nikos Athanasiou, Elias Iosif and Alexandros Potamianos. 2018. Neural activation semantic models: compositional lexical semantic models of localized neural activations. In *Proceedings of the 27th International Conference on Computational Linguistics*, 2867–2878, Association for Computational Linguistics, Santa Fe, New Mexico, USA.

Marco Baroni and Alessandro Lenci 2010. Distributional Memory: A General Framework for Corpus-Based Semantics. In *Computational Linguistics*, 36: 673–721.

Pier Marco Bertinetto. 1986. Tempo, aspetto e azione nel verbo italiano, Firenze, Accademia della Crusca.

Lucia Busso, Ludovica Pannitto e Alessandro Lenci. 2018. Modelling the Meaning of Argument Constructions with Distributional Semantics. Are constructions enough? In *Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it 2018*.

Alan Cruse. 1973. Some Thoughts on Agentivity. In *Journal of Linguistics*, 9(1):11-23.

Jacob Devlin et al. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *CoRR, Vol. abs/1810.04805.*

Guy Emerson. 2020. What are the Goals of Distributional Semantics? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1-18.

Allyson Ettinger. 2019. What {BERT} is not: Lessons from a new suite of psycholinguistic diagnostics for language models. In *CoRR, Vol. abs/1907.13528.*

Cécile Fabre and Alessandro Lenci. 2015. Distributional Semantics Today. Introduction to the special issue. In *Traitement automatique des langues. Sémantique distributionelle (ATALA)*, 50(2) :7-20.

Scott Grimm. 2018. Grammatical number and the scale of individuation. In *Language*, 94:527-594.

Alessandro Lenci e Alessandra Zarcone. 2008. Computational Models of Event Type Classification in Context. In *Language Resources and Evaluation Conference*.

Alessandro Lenci. 2008. Distributional semantics in linguistic and cognitive research. *In The Italian Journal of Linguistics*, 20:1-20.

Yang Liu et al. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach, In *CoRR, Vol. abs/1907.11692*.

Timothee Mickus et al. 2020. What do you mean, BERT? Assessing BERT as a Distributional Semantics Model. In *Proceedings of the Society for Computation in Linguistics*, 3(34):1-12.

Giulio Ravasio and Leonardo Di Perna. GilBERTo: An Italian pretrained language model based on RoBERTa. DOI: https://github.com/idb-ita/GilBERTo.

Vered Shwartz and Ido Dagan. 2019. Still a Pain in the Neck: Evaluating Text Representations on Lexical Composition. *In CoRR, Vol. abs/1902.10618.*

Ian Tenney et al. 2019. What do you learn from context? Probing for sentence structure in contextualized word representations. In *CoRR, Vol. abs/1905.06316.*

Peter D. Turney. 2006. Expressing Implicit Semantic Relations without Supervision. In *CoRR, Vol abs/cs/0607120.*

Robert D. van Valin and Randy J. LaPolla. 1997. Syntax: Structure, Meaning, and Function. In *Linguistics, Cambridge University Press*.

Zeno Vendler. 1967. Causal relations. In *Journal of Philosophy*, 64:704-713.

Koki Washio and Tsuneaki Kato. 2018. Neural Latent Relational Analysis to capture lexical semantic relation in a vector space. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing,* 594-600.

Mo Yu and Mark Dredze. 2014. Improving Lexical Embeddings with Semantic Knowledge. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 2:545-550.

Xunjie Zhu, Tingfeng Li and Gerard de Melo. 2018. Exploring Semantic Properties of Sentence Embeddings, In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2:632-637.

Investigating Continued Pretraining for Zero-Shot Cross-Lingual Spoken Language Understanding

Samuel Louvan[1,2], **Silvia Casola**[1], **Bernardo Magnini**[1]
1. Fondazione Bruno Kessler, Italy
2. University of Trento, Italy
slouvan@fbk.eu, scasola@fbk.eu, magnini@fbk.eu

Abstract

Spoken Language Understanding (SLU) in task-oriented dialogue systems involves both intent classification (IC) and slot filling (SF) tasks. The *de facto* method for zero-shot cross-lingual SLU consists of fine-tuning a pretrained multilingual model on English labeled data before evaluating the model on unseen languages. However, recent studies show that adding a second pretraining stage (*continued pretraining*) can improve performance in certain settings. This paper investigates the effectiveness of continued pretraining on unlabeled spoken language data for zero-shot cross-lingual SLU. We demonstrate that this relatively simple approach benefits either SF and IC task across 8 target languages, especially the ones written in Latin script. We also find that discrepancy between languages used during pretraining and fine-tuning may introduce training instability, which can be alleviated through code-switching.

1 Introduction

In task-oriented dialogue systems, a Spoken Language Understanding (SLU) component typically involves intent classification (IC) and slot filling (SF) (Tur and De Mori, 2011) tasks. For example, in "*Show me the fares for Delta flights from Dallas to San Francisco*", the intent is ASKING AN AIRFARE and its corresponding slots are *Delta* (AIRLINE-NAME), *Dallas* (CITY-ORIGIN), and *San Francisco* (CITY-DESTINATION). Scaling SLU models to other languages is still challenging, especially when there is limited or no labeled data available in the target language (Louvan and Magnini, 2020).

To approach this problem, previous work studies IC and SF tasks in a zero-shot cross-lingual setting (Schuster et al., 2019; Upadhyay et al., 2018; Xu et al., 2020), where it is assumed that a labeled dataset is only available for a high resource language (e.g., English). With the rise of pretrained multilingual language models (LMs) (Devlin et al., 2019; Lample and Conneau, 2019) the most common approach is by fine-tuning the pretrained multilingual model on the English labeled data, and then evaluate the model directly on the target language data that are not seen during fine-tuning.

While direct fine-tuning serves as a strong baseline, pretrained LMs are not necessarily *universal* and they may need domain-specific adaptation. Recent works have shown that adding a second pretraining stage (or *continued pretraining*) before fine-tuning can give positive impact on the model performance (Beltagy et al., 2019; Lee et al., 2020; Gururangan et al., 2020). During continued pretraining, we continue training the pretrained language model using a *domain-specific* or *task-specific* unlabeled dataset, with the same masked language model objective. This stage is useful to alleviate the *domain mismatch* between the original pretraining and the target task data. By continued pretraining on domain specific unlabeled data, the model acquires prior knowledge which is expected to be helpful in the fine-tuning stage. This approach has shown promising results on text classification, typically on English. However, it remains unclear whether it is applicable in the context of *zero-shot cross-lingual* SLU.

In contrast to previous work which has mostly focused on English classification tasks, we investigate the effectiveness of continued pretraining for zero-shot cross-lingual SLU tasks on eight target languages. Our study reveals that the existing continued pretraining method (Gururangan et al.,

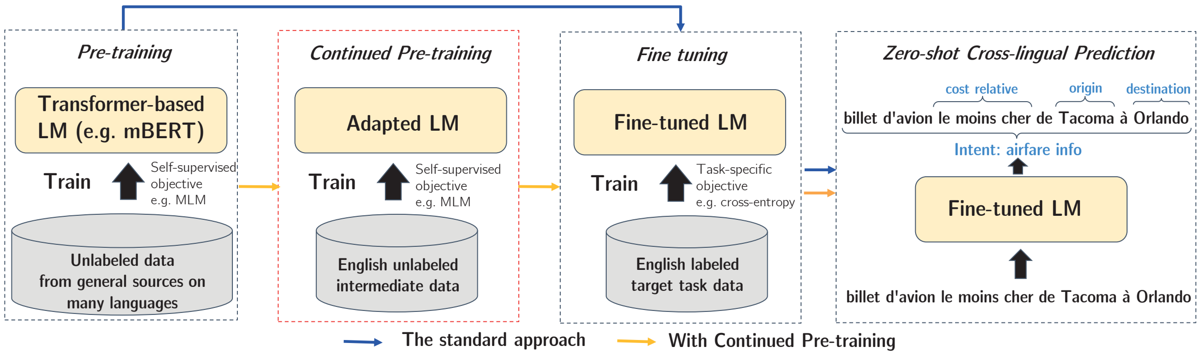


Figure 1: The overall stages of zero-shot cross lingual SLU using a pretrained multilingual model. The standard approach follows the stages marked with blue arrows (*direct fine-tuning*). We investigate the effectiveness of adding a continued pretraining stage (red dashed box) in the overall pipeline.

2020), that is successful in English text classification tasks, does not always generalize to the context of zero-shot cross-lingual SLU. We focus on the following research questions:

(Q1) *Is continued pretraining effective for zero-shot cross-lingual SLU tasks?*
↪ Our experiments on the MultiATIS++ dataset (Xu et al., 2020) reveal that incorporating continued pretraining on intermediate English data can improve performance over direct fine-tuning for all languages, on zero-shot SLU. The performance gain is especially evident for languages with Latin script writing system. The benefit of continued pretraining diminishes as we inject cross-lingual supervision in the fine-tuning stage, even with simple data augmentation through code-switching.

(Q2) *What are the factors that influence the effectiveness of the continued pretraining stage?*
↪ Using the target language for continued pretraining before fine-tuning on English can be detrimental to the overall performance. However, this can be largely alleviated by code-switching the fine-tuning data. We also observe that performance improvement are not obtained by merely adding more continued pretraining data; higher domain similarity between the continued pretraining data and the fine-tuning data is indeed more important.

2 Continued Pretraining in Zero-Shot SLU

Figure 1 shows a comparison between the standard direct fine-tuning approach with the continued pretraining approach. The main difference is the additional intermediate pretraining stage (second block in Figure 1), in which we continue training the model on an intermediate unlabeled data using the same masked language modeling objective. As the original pretraining data is relatively far from the task-oriented dialogues used in SLU, we hypothesize that continued pretraining can alleviate the *domain* mismatch and ingest a better prior knowledge that will be useful during fine-tuning.

Intermediate Data for Continued Pretraining. We define several criteria for the intermediate pretraining data for the continued pretraining stage. First, their domain should be relatively close to the target dataset. We interpret the term domain as a multidimensional *variety space* (Ramponi and Plank, 2020; Plank, 2016): a domain comprises multiple aspects (style, topic, and genre (van der Wees et al., 2015)) that contribute to the text variation. Using this perspective and considering the target domain of a task-oriented dialogue system, we require that the intermediate data comprises text that presents a *spoken language dialog* style and covers a *broad range of topics*. Second, the dataset should be several magnitudes larger in size than the target task dataset. Finally, it must be available in many languages to support our study of continued pretraining with the target language.

3 Experimental Setup

In this section, we describe the experimental settings related to models, evaluation metrics, and datasets.

3.1 Models

For all of our experiments, we use a transformer-based model (Vaswani et al., 2017), namely multilingual BERT (mBERT) (Devlin et al., 2019), as the pretrained model. This model was pretrained on Wikipedia articles covering 104 languages, and we use the *bert-base-multilingual-cased* version.

Continued Pretraining. For the continued pretraining stage, we further train mBERT with unlabeled intermediate data using only the Masked Language Modeling (MLM) objective for 12.5K steps, and mostly adopt the hyperparameters in Gururangan et al. (2020). We compare the following configurations: (i) DAPT_{Tgt} a continued domain adaptive pretraining (DAPT) of mBERT on intermediate unlabeled data on the target language. (ii) DAPT_{En} a continued DAPT of mBERT on intermediate unlabeled data on English.

Fine-Tuning. As the baseline model, without any adaptation (No DAPT), we use the joint IC and SF model architecture (Chen et al., 2019). This model is the state-of-the-art for IC and SF (Louvan and Magnini, 2020), and it is often used as one of the baselines in recent zero-shot cross-lingual SLU studies (Xu et al., 2020; Li et al., 2021). The model is trained on the English dataset; as the setup is zero-shot cross-lingual and we use the model's last epoch for zero-shot evaluation following Xu et al. (2020). We evaluate the effectiveness of each of the DAPT configurations when applied to the following fine-tuning scenarios:

- Fine-tuning on English (FINETUNE-EN). This is the standard fine-tuning scenario, where we take mBERT either with DAPT or no DAPT, fine-tune it on the English IC and SF data, and then perform zero-shot prediction to all target language data.
- Fine-tuning on the English *code-switched* data (FINETUNE-CS). In this scenario, we perform data augmentation on the English fine-tuning dataset via code-switching. We follow the approach from Qin et al. (2020), where we replace the English words with their translation in the target language using the Panlex bilingual dictionary (Kamholz et al., 2014). Given a training batch, we randomly sample sentences and tokens to replace. We use the same hyperparameter used by Qin et al. (2020), that defines both sentence and word ratio to control the word replacement. We include FINETUNE-CS because we want to study the benefits of DAPT when adding stronger cross-lingual supervision in the fine-tuning stage.

We did not experiment with more complex models models as our main goal is to investigate the effect of the the continued pretraining stage, rather than achieving the state of the art performance per se.

Implementation & Model Evaluation metric. For the intent and SF models, we adapt the implementation from Qin et al. (2020) in which they make it publicly available (https://github.com/kodenii/CoSDA-ML). The sentence and token ratio replacement for code-switching is set to 1.0 and 0.9 respectively. For training, the learning rate is set to 10^{-5}, batch size is set to 32, number of epoch is set to 20. We did not do extensive hyperparameter tuning, as this is a zero-shot cross lingual case where the target dataset is not available, we use the same hyperparameters as Xu et al. (2020). For the continued pretraining we use the language modeling script from Huggingface (Wolf et al., 2019). We use the `bert-base-multilingual-cased`, hidden state size is 768, we apply dropout probability of 0.1. The number of training steps is 12,500 following Gururangan et al. (2020), the batch size is set to 16.

3.2 Dataset

SF and IC Dataset. We use the MultiATIS++ (Xu et al., 2020) dataset, which contains nine languages (Table 1). The dataset is derived from the original ATIS English dataset (Hemphill et al., 1990), widely used as a benchmark for IC and SF for task-oriented dialogue systems. Utterances are related to conversations of a user asking for flight information to a system.

Language	#train / #dev /#test	#slot	#intent
English (EN)	4.4K/ 490 / 893	83	24
German (DE)	4.4K / 490 / 892	83	24
Spanish (ES)	4.4K/ 490 / 893	83	24
French (FR)	4.4K / 490 / 893	83	24
Portuguese (PT)	4.4K / 489 / 892	83	24
Hindi (HI)	1.4K / 160 / 888	74	22
Japanese (JA)	4.4K / 490/ 886	83	24
Chinese (ZH)	4.4K / 490 / 893	83	24
Turkish (TR)	0.6K / 60/ 715	70	21

Table 1: Multi-ATIS++ (Xu et al., 2020) statistics.

Continued Pretraining Dataset. We use the OpenSubtitle (OpenSub) (Lison and Tiedemann, 2016) (Table 2) dataset for the continued pretraining stage for several reasons. First, the dataset is constructed from movies and TV series containing *spoken language* in dialogue settings covering a broad range of topics. Second, OpenSubtitle covers all the *languages* that we use on the downstream tasks, which enables us to evaluate not only

DAPT_{En} but also DAPT_{Tgt}. Third, the dataset is large in size, thus ideal for continued pretraining. Typically, the dataset used for continued pretraining is larger than that used for fine-tuning. For our experiments we randomly sampled 100K sentences for each language in the OpenSub dataset, resulting in a dataset around 20 times larger than the downstream task dataset.

Language	Total Tokens
English (EN)	734,302
German (DE)	691,039
Spanish (ES)	711,264
French (FR)	739,551
Portuguese (PT)	676,789
Hindi (HI)	688,675
Japanese (JA)	747,780
Chinese (ZH)	611,700
Turkish (TR)	554,709

Table 2: OpenSub (Lison and Tiedemann, 2016) dataset statistics. Each language has 100 K utterances.

4 Results

The main goal of our experiment is to answer research question (**Q1**). Table 3 compares the zero-shot performance for SF and IC across languages. In terms of language (by column in Table 3), we observe that all languages improve over No-DAPT in at least one DAPT setting, suggesting that DAPT is effective across languages. Observing the results per task, SF benefits from either DAPT_{En} or DAPT_{Tgt} for German, Spanish, French, Portuguese, and Turkish, which all are languages with Latin scripts writing system. For these languages, the margin obtained from DAPT when fine-tuning on English (FINETUNE-EN) is higher than when we apply DAPT on code-switched data (FINETUNE-CS). The margin of DAPT when applied on FINETUNE-CS diminishes because FINETUNE-CS uses a stronger supervision signal in the fine-tuning stage, thus providing a higher baseline. For languages with non-Latin script writing system, continued pretraining is less useful; we only observe marginal improvement on Japanese when applying DAPT_{En} and FINETUNE-EN. Similar to Lauscher et al. (2020), we believe that performance is also affected by typological language proximity such as the subject, verb, and object ordering, phonology features or other aspect related to the original size of the pre-training data of mBERT. We leave this for future work.

DAPT is less effective for IC than for SF. The only language that consistently benefits from continued pretraining in both fine-tuning scenarios is Turkish. We found that it is harder to improve the model performance of languages with Latin script through DAPT because the baseline is relatively high; a stronger supervision signal would thus be needed. The performance gain is small even for those languages that do benefit from DAPT. We also observe that using a different language between continued pretraining and fine-tuning stages, DAPT_{Tgt} and FINETUNE-EN, may hamper performance.

5 Analysis and Discussion

To answer the research question (**Q2**), we analyze our results focusing on the performance variation when using different languages in DAPT and fine-tuning (§5.1) and the effect of domain distribution in different sources for DAPT_{En} (§5.2).

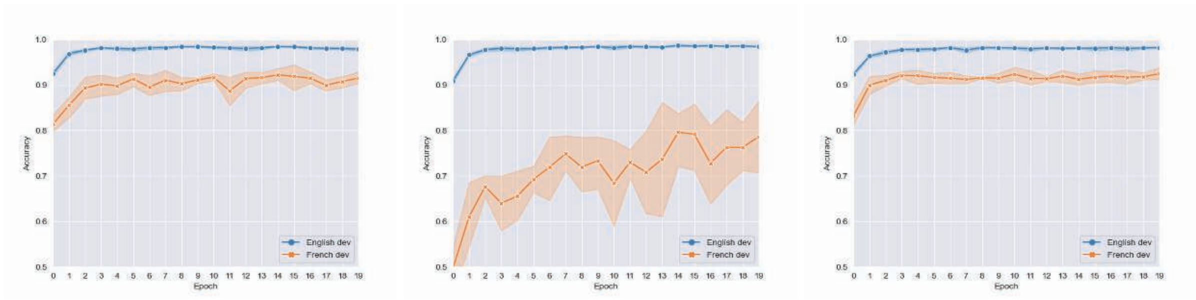
5.1 Performance Variation when Applying DAPT

As we have noticed in Section §4, there are cases where performance drop when we use DAPT_{Tgt} and FINETUNE-EN, especially for IC. This behaviour holds even for languages relatively close to English, such as German and French. One possible reason for the drop in accuracy is that the language difference introduces instability in fine-tuning. Our post-hoc analysis shows that the target language performance during training on the dev set has a large deviation and continues fluctuating even after the English dev performance has stabilized. This observation resonates with a previous study from Keung et al. (2020), which shows that, for zero-shot text classification, English dev performance often does not correlate with those of the target language. Using DAPT_{Tgt} and FINETUNE-EN pronounces the disagreement of performance between the English and the target dev set. Figure 2 shows the comparison of the IC performance during training across continued pretraining strategies when fine-tuning on English for French. However, for the SF task, we do not observe a large performance variation even with a language mismatch: this might indicate that text classification is more susceptible to instability than sequence tagging. The variability caused by DAPT_{Tgt} is largely alleviated when we use DAPT_{En}. For the FINETUNE-CS scenario, the system is relatively stable even when combined

SF F1								
	DE	ES	FR	PT	HI	JA	ZH	TR
FINETUNE-EN								
No-DAPT	65.3	71.3	64.0	61.9	47.5	62.2	66.3	27.4
$\Delta\text{DAPT}_{\text{Tgt}}$	+4.0	−2.4	−7.7	−0.6	−12.9	−9.7	−0.6	+18.5
$\Delta\text{DAPT}_{\text{En}}$	+2.1	+0.9	+5.9	+1.4	−4.5	+0.8	−0.2	−5.8
FINETUNE-CS								
No-DAPT	75.5	80.8	71.9	72.0	58.1	67.1	81.6	72.0
$\Delta\text{DAPT}_{\text{Tgt}}$	−0.2	−0.4	+0.5	+1.1	−3.9	−6.3	−1.2	−10.9
$\Delta\text{DAPT}_{\text{En}}$	+0.4	+0.1	+4.6	+1.2	−13.9	−8.4	−0.7	−15.8

IC ACCURACY								
	DE	ES	FR	PT	HI	JA	ZH	TR
FINETUNE-EN								
No-DAPT	90.0	91.9	92.1	92.8	81.1	83.0	87.1	61.2
$\Delta\text{DAPT}_{\text{Tgt}}$	−10.8	+0.5	−13.3	−1.6	−13.3	−1.9	−2.9	+8.1
$\Delta\text{DAPT}_{\text{En}}$	−0.8	−0.1	+0.1	−0.6	−2.5	−0.5	−2.4	+8.3
FINETUNE-CS								
No DAPT	95.1	96.4	96.6	94.2	85.6	85.1	88.0	66.2
$\Delta\text{DAPT}_{\text{Tgt}}$	−1.1	−0.2	−0.5	+1.3	+0.6	−2.4	+0.3	+3.9
$\Delta\text{DAPT}_{\text{En}}$	−1.6	−0.2	−0.2	+0.4	−0.8	−2.6	−7.3	+12.3

Table 3: Performance comparison on the test set for SF and IC. Scores for No DAPT are the average slot F1 and intent accuracy over five runs. The $\Delta\text{DAPT}_{\text{Tgt}}$ and $\Delta\text{DAPT}_{\text{En}}$ indicate the delta between DAPT and No DAPT.



(a) No DAPT + FINETUNE-EN (b) DAPT_{Tgt}+ FINETUNE-EN (c) DAPT_{En}+ FINETUNE-EN

Figure 2: Post-hoc analysis: *development set* performance variation on IC between English and French, using FINETUNE-EN and applying different DAPT strategies.

with DAPT_{Tgt} or DAPT_{En}.

5.2 Domain Relevance for DAPT_{En}

We aim at investigating whether the improvement from the continued pretraining comes from the domain relevance of the intermediate data. For this purpose, we selected a few *written text* datasets instead of spoken language, which are focused on a *specific topic*. Specifically, we use the European Medicines Agency (EMEA) and European Central Bank corpus (ECB) from Tiedemann (2012). EMEA contains articles about human, veterinary, or herbal medicines extracted from the EMEA website. ECB contains financial documents that are extracted from the website and documentation of the European Central Bank. In order to check that EMEA and ECB are more distant in terms of domain from MultiATIS than OpenSub, we compute the Jensen Shannon Divergence (JSD) measure of the term distribution (Dai et al., 2020; Ruder and Plank, 2017). We compute the JSD between the MultiATIS English dataset that is used for fine-tuning and each English intermediate dataset. Based on the JSD measure, EMEA and ECB are more distant to MultiATIS than OpenSub (Table 4).

For each intermediate dataset, we randomly sample 100K sentences and use them for continued pretraining. We compare the SF performance of DAPT_{En} with FINETUNE-EN on Open-

	OpenSub	EMEA	ECB
JSD	0.419	0.391	0.397

Table 4: Domain similarity between MultiATIS and each of the intermediate data.

Lang.	No DAPT	ΔDAPT$_{\mathrm{En}}$		
		OpenSub	EMEA	ECB
DE	65.3	+2.1	−2.5	−9.5
ES	71.3	+0.9	+0.9	+1.3
FR	64.0	+5.9	+2.0	+0.7
PT	61.9	+1.4	−0.3	−9.1
Avg		+2.5	+0.005	−4.1

Table 5: Comparison of SF performance with different intermediate data.

Sub, EMEA, and ECB in Table 5. We focus on languages that belongs to Indo-European family which mostly obtain benefit from DAPT on SF (Table 3) Overall, we see that DAPT using OpenSub obtains improvements over No-DAPTin all cases. The DAPT performance using EMEA and ECB are lower than OpenSub in most cases. Even for DE and PT languages, DAPT with ECB obtains substantially lower performance than No-DAPT. However, there are cases when EMEA or ECB match or even perform better than OpenSub i.e., for Spanish. These cases indicate that performing *data selection* before continued pretraining could be beneficial to construct more optimal DAPT dataset. It would be interesting also to observe how continued pre-training would work using smaller unlabeled pre-training data but more task relevant. We leave this possibility for future work.

6 Related Work

Zero-Shot Cross-Lingual SLU. Before the advent of the pre-trained multilingual transformer models, most approaches relied on pre-trained cross-lingual embeddings to perform zero-shot SLU. Upadhyay et al. (2018) uses cross-lingual embedding (Bojanowski et al., 2017) to perform zero-shot SLU while Schuster et al. (2019) uses multilingual embedding (Cove) from pre-trained multilingual bi-LSTM encoder used in Neural Machine Translation (NMT). Liu et al. (2019) leverages transferable latent variables to improve the sentence representation across languages. More recently, as pre-trained multilingual transformer models show potential in zero-shot settings, most approaches focus on improving their multilingual representation through augmentation and alignment methods. Qin et al. (2020) proposes multilingual code-switching using a bi-lingual dictionary to improve mBERT's multilingual representation. Xu et al. (2020) introduces soft alignment of slots between English and the target language produced by a machine translation system that eliminates the need for an annotation projection pipeline. Kulshreshtha et al. (2020) study the effect of various cross-lingual alignment methods to improve mBERT representation.

Continued Pre-training Domain adaptation is a long-studied problem in the NLP community (Daumé III, 2007; Blitzer et al., 2007), in which we assume data in the target domain might be hard to obtain while being abundant in source domains. Continued pre-training – where the model is trained on relevant data using the same pre-training objective – is used for mitigating the distribution mismatch between the pre-training and the fine-tuning data in terms of *domain* (Logeswaran et al., 2019; Han and Eisenstein, 2019; Gururangan et al., 2020; Beltagy et al., 2019), *task* (Gururangan et al., 2020), and *language* (Pfeiffer et al., 2020). A complementary approach performs a first fine-tuning on related auxiliary tasks (for which training data are easy to obtain) before the final fine-tuning on the downstream task (Arase and Tsujii, 2019; Garg et al., 2020; Khashabi et al., 2020). Our work is in line with Gururangan et al. (2020) where we investigate further the effectiveness of continued pre-training in the context of zero-shot cross-lingual SLU.

7 Conclusion

We systematically study the effectiveness of continued pre-training of a multilingual model on intermediate English unlabeled spoken language data for zero-shot cross-lingual tasks, namely intent classification and slot filling, on 8 languages. Our results show that the domain knowledge learned in English is transferable to other languages. The gain from continued pre-training diminishes as we inject cross-lingual supervision in the fine-tuning stage. There are several factors that influence the effectiveness of the continued pre-training: (i) Using different language between pre-training and fine-tuning can hamper performance and introduce instability in the model training, which can be alleviated with code switching. (ii) Domain similarity is important. The more similar – in terms of data distribution – the intermediate data to the target dataset yields better performance.

References

Yuki Arase and Jun'ichi Tsujii. 2019. Transfer fine-tuning: A BERT case study. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 5393–5404, Hong Kong, China, November. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 3615–3620, Hong Kong, China, November. Association for Computational Linguistics.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 440–447, Prague, Czech Republic, June. Association for Computational Linguistics.

P. Bojanowski, E. Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5:135–146.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. BERT for joint intent classification and slot filling. ArXiv, abs/1902.10909.

Xiang Dai, Sarvnaz Karimi, Ben Hachey, and Cecile Paris. 2020. Cost-effective selection of pretraining data: A case study of pretraining BERT on social media. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1675–1681, Online, November. Association for Computational Linguistics.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 256–263, Prague, Czech Republic, June. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), pages 4171–4186. Association for Computational Linguistics.

Siddhant Garg, Thuy Vu, and Alessandro Moschitti. 2020. Tanda: Transfer and adapt pre-trained transf ormer models for answer sentence selection. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05):7780–7788, Apr.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8342–8360, Online, July. Association for Computational Linguistics.

Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 4238–4248, Hong Kong, China, November. Association for Computational Linguistics.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, USA, June 24-27, 1990. Morgan Kaufmann.

David Kamholz, Jonathan Pool, and Susan Colowick. 2014. PanLex: Building a resource for panlingual lexical translation. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14), pages 3145–3150, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

Phillip Keung, Y. Lu, Julian Salazar, and Vikas Bhardwaj. 2020. Don't Use English Dev: On the Zero-Shot Cross-Lingual Evaluation of Contextual Embeddings. In EMNLP.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1896–1907, Online, November. Association for Computational Linguistics.

Saurabh Kulshreshtha, José Luis Redondo García, and Ching-Yun Chang. 2020. Cross-lingual alignment methods for multilingual BERT: A comparative study. In EMNLP.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. Advances in Neural Information Processing Systems (NeurIPS).

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4483–4499, Online, November. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, D. Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics, 36:1234 – 1240.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 2950–2962, Online, April. Association for Computational Linguistics.

Pierre Lison and J. Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles. In LREC.

Zihan Liu, Jamin Shin, Yan Xu, Genta Indra Winata, Peng Xu, Andrea Madotto, and Pascale Fung. 2019. Zero-shot cross-lingual dialogue systems with transferable latent variables. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 1297–1303, Hong Kong, China, November. Association for Computational Linguistics.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3449–3460, Florence, Italy, July. Association for Computational Linguistics.

Samuel Louvan and Bernardo Magnini. 2020. Recent neural methods on slot filling and intent classification for task-oriented dialogue systems: A survey. In Proceedings of the 28th International Conference on Computational Linguistics, pages 480–496, Barcelona, Spain (Online), December. International Committee on Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7654–7673, Online, November. Association for Computational Linguistics.

Barbara Plank. 2016. What to do about non-standard (or non-canonical) language in NLP. arXiv preprint arXiv:1608.07836.

L. Qin, Minheng Ni, Y. Zhang, and W. Che. 2020. Cosda-ml: Multi-lingual code-switching data augmentation for zero-shot cross-lingual NLP. In IJCAI.

Alan Ramponi and Barbara Plank. 2020. Neural unsupervised domain adaptation in NLP—A survey. In Proceedings of the 28th International Conference on Computational Linguistics, pages 6838–6855, Barcelona, Spain (Online), December. International Committee on Computational Linguistics.

Sebastian Ruder and Barbara Plank. 2017. Learning to select data for transfer learning with Bayesian optimization. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 372–382, Copenhagen, Denmark, September. Association for Computational Linguistics.

Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019. Cross-lingual transfer learning for multilingual task oriented dialog. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), pages 3795–3805. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Ugur Dogan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12), Istanbul, Turkey, may. European Language Resources Association (ELRA).

Gokhan Tur and Renato De Mori. 2011. Spoken language understanding: Systems for extracting semantic information from speech. John Wiley & Sons.

Shyam Upadhyay, Manaal Faruqui, Gökhan Tür, Dilek Z. Hakkani-Tür, and Larry Heck. 2018. (almost) zero-shot cross-lingual spoken language understanding. 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 6034–6038.

Marlies van der Wees, Arianna Bisazza, Wouter Weerkamp, and Christof Monz. 2015. What's in a domain? Analyzing genre and topic differences in statistical machine translation. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 560–566, Beijing, China, July. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman

Garnett, editors, Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. ArXiv, abs/1910.03771.

Weijia Xu, Batool Haider, and Saab Mansour. 2020. End-to-end slot alignment and recognition for cross-lingual NLU. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 5052–5063, Online, November. Association for Computational Linguistics.

Linking the Lewis & Short Dictionary to the LiLa Knowledge Base of Interoperable Linguistic Resources for Latin

Francesco Mambrini, Eleonora Litta, Marco Passarotti, Paolo Ruffolo
CIRCSE Research Centre
Università Cattolica del Sacro Cuore
Largo Gemelli, 1 - 20123 Milan, Italy
francesco.mambrini@unicatt.it, eleonoramaria.litta@unicatt.it,
marco.passarotti@unicatt.it, paolo.ruffolo@posteo.eu

Abstract

This paper describes the steps taken to include data from the Lewis & Short bilingual Latin-English dictionary into the Knowledge Base of linguistic resources for Latin LiLa. First, data were extracted from the original XML and matched with entries in LiLa, overcoming ambiguities and structural inconsistencies in the source. Subsequently, senses were modelled using the Ontolex Lemon Lexicographic module (lexicog), so that they could be included in the LiLa Knowledge Base and thus made interoperable with the (meta)data of the linguistic resources for Latin therein interlinked.

1 Introduction

Since the pioneering times of 1949, when the Jesuit Roberto Busa persuaded Thomas Watson Sr., CEO of IBM, to fund his project aimed at processing the Latin texts of Thomas Aquinas with computers (Jones, 2016), scholars in the areas of Computational Linguistics, Literary Computing and Digital Humanities have built a plethora of linguistic resources for both modern and historical languages.

Particularly over the last two decades, many and diverse linguistic resources have been made available for Latin. These consist in corpora of texts spanning different eras and genres[1], dependency treebanks[2] and lexica[3]. These digital resources join the large set of textual and lexical resources that were created over the centuries for Latin: textual collections, thesauri, lexica, glossaries and mono/bilingual dictionaries. Among the latter, we could mention, for instance, the *Oxford Latin Dictionary* (Glare, 1968), the *Dictionary of medieval Latin from British sources* (Ashdowne et al., 1975), the Forcellini lexicon (Forcellini and Facciolati, 1871) and the still under construction *Thesaurus Linguae Latinae* (Ehlers, 1968), many of which are today accessible also in digital format.

However, the impact of these digital resources on the everyday work of classicists is still limited. On the one side, this is due to the still existing divisive dichotomy between "traditional" Humanities and computational approaches. On the other, it is a matter of fact that classicists are not yet put in the best condition to fully exploit all available resources for ancient languages, as these are currently scattered across the web in uncommunicative blocks, using different query languages, data formats, annotation criteria and tagsets. The last decade has seen a number of exploratory solutions to tackle the sparseness of linguistic resources. Among them, the European infrastructure CLARIN[4] represents a common hub where data and metadata of resources collected in single repositories (at national level) can be searched (through the so-called Virtual Language Observatory) and processed with different tools (through the CLARIN Language Resource Switchboard). As for Classical languages, *Logeion*[5] is a meta-

[1]See, for example, *Musisque deoque* for Classical Latin poetry (Manca et al., 2011), *CLaSSES*, containing epigraphic material (De Felice et al., 2015), the large corpus of Classical Latin prose and poetic texts by LASLA (Denooz, 2007) and *CroALa*, which brings together writings by Croatian authors produced between the 10th and 20th centuries (Jovanović, 2012).

[2]*Index Thomisticus* Treebank (Passarotti, 2019), Late Latin Charter Treebank (Cecchini et al., 2020a), UDante (Cecchini et al., 2020b), PROIEL (Eckhoff et al., 2018) and Latin Dependency Treebank (Bamman and Crane, 2011).

[3]Such as, for instance, valency and subcategorisation lexica (Passarotti et al., 2016; McGillivray and Vatri, 2015), the Latin WordNet (Minozzi, 2017) and word lists (Tombeur, 1998; Ramminger, 2008).

[4]https://www.clarin.eu.

[5]https://logeion.uchicago.edu/lexidium.

dictionary that allows to query together the lexical entries of several dictionaries for Ancient Greek and Latin, while *Corpus Corporum*[6] is a meta-collection that allows searches across more than twenty different corpora for Latin. However, what such initiatives still lack is to provide a real interoperability between distributed resources, which would result in interaction at both syntactic (structural) and semantic (conceptual) level.

Syntactic interoperability is defined as 'the ability of different systems to process (read) exchanged data either directly or via trivial conversion', using a common data model consisting of shared protocols and data formats. Semantic interoperability, on the other hand, is 'the ability to automatically interpret exchanged information meaningfully and accurately in order to produce useful results', by using a set of common linguistic data categories defined in *ad-hoc* ontologies (Ide and Pustejovsky, 2010).

Attaining syntactic and semantic interoperability between distributed linguistic resources is the objective of the Linguistic Linked Open Data (LLOD) community, which applies the principles of the Linked Data paradigm (Bizer et al., 2008) to the (meta)data contained in linguistic resources. As for Classical languages, the LiLa Knowledge Base (KB)[7] (Passarotti et al., 2020) makes textual and lexical resources for Latin interact through a commonly used data model, called the *Resource Description Framework* (RDF) (Lassila et al., 1998), and ontologies developed and shared by the LLOD community. In this way, the linked resources become interoperable with each other as well as with those for other languages described following the same structural and conceptual principles.

Based on a large collection of "canonical forms" (lemmas) - the so-called "Lemma Bank", LiLa achieves interoperability between resources by linking all those entries in lexical resources and tokens in corpora that point to the same lemma in the LiLa collection.

The lexical resources for Latin linked so far to LiLa include a word formation lexicon (Pellegrini et al., 2021), a polarity lexicon (Sprugnoli et al., 2020), an etymological dictionary (Mambrini and Passarotti, 2020) and a joint resource providing a manually checked subset of the Latin WordNet and a valency lexicon (Mambrini et al., 2021). The most recent among the LiLa connections is the bilingual Latin-English dictionary by Charlton Lewis and Charles Short (1879). The inclusion of this type of lexicon in LiLa was much needed, as no resource providing semantic information consisting of translations and definitions was available in the network of connected resources before. Since Lewis & Short is the first lexical resource of its kind included in LiLa, the process of its linking to the KB opened a number of LLOD-related challenges.

This paper describes how such challenges have been tackled and is organised as follows: Section 2 describes the Lewis & Short dictionary in its main characteristics. Section 3 discusses the ontologies involved in the modelling phase, the challenges that need to be overcome in the representation of the linguistic data as LLOD (3.1), and the strategies adopted to represent the dictionary entries using the chosen vocabularies (3.2). Finally, Section 4 discusses conclusions and highlights directions for future work.

2 The "Lewis & Short" Dictionary

2.1 The Printed and Digital Dictionary

The *Latin Dictionary*, curated by Ch. T. Lewis and Ch. Short and commonly referred to as the "Lewis & Short" (L&S), was published by Harper and Oxford University Press in 1879 (Lewis and Short, 1879). Though based on previous work by German scholars, it remained a standard in Latin lexicography in the English-speaking world until it was superseded by the *Oxford Latin Dictionary* (Glare, 1968).

In the digital age, its importance rests on two grounds. On the one hand, its relevance for the history of Classical Scholarship is undeniable. On the other hand, also on account of its copyright status, as the dictionary belongs now to the public domain, the L&S has quickly become one of the most used and best curated digital Latin dictionaries on the web. Following the same workflow used for the *Greek-English Lexicon* (Liddell et al., 1940), the Perseus Project has developed a widely used digital edition of the dictionary based on the standards of the Text Encoding Initiative (TEI) (Rydberg-Cox, 2002). The digital L&S has been incorporated in the word-search tools available on the Perseus website and in a series of other

[6] http://www.mlat.uzh.ch/MLS/.
[7] https://lila-erc.eu.

desktop and web applications.[8]

Perseus' TEI edition is the point of departure of our work.[9] Though its publication was a remarkable achievement, this electronic text is not exempt from occasional flaws and inconsistencies, which had to be taken into account.

In the digital edition, entries from the L&S are based on an XML encoding of the whole dictionary. The XML structure, albeit not always consistent, offers the following information about each word:

1. Entry: the headword. Entries are encoded within the TEI element `<entryFree>` and are 51,596 in total.[10]

2. Information about inflection, encoded as attributes in the XML and visualised in the output reproducing the customary descriptions for Latin dictionaries, e.g. a masculine noun of the second declension (e.g. *gallus* 'cock') is followed by the genitive singular ending of the word ('i'), and the abbreviation for gender 'm.' (e.g. *gallus, i*, m.).

3. Etymological or derivational information, encoded within the same element `<etym>`.

4. Sense(s): these act as containers where the meaning of the word is matched with a number of representative citations from Classical Latin sources. Each citation is accompanied by its canonical reference (e.g. "Cic. Sen. 8, 26" for a reference to Cicero, *De Senectute*, chapter 8, paragraph 26).

Entries can contain what we call "sub-entries", words that are not given a record of their own, but are discussed within another entry. Usually, these sub-entries consist of lexicalised present and past participles like, for example, *adolescens* 'young man' – sub-entry of *adolesco* 'to grow up'; another instance is the substantivised forms of adjectives, such as *verum* 'the truth' – sub-entry of *verus* 'true'. Sub-entries are encoded within the `<sense>` element and followed by the same type of inflectional information structured as the main entries.

[8]One example is the app *Diogenes* for querying corpora of Greek and Latin texts: https://d.iogen.es/.

[9]The digital edition is available from the repository of the Perseus DL and is distributed under a CC BY SA 4.0 license: https://github.com/PerseusDL/lexica.

[10]See https://tei-c.org/release/doc/tei-p5-doc/en/html/ref-entryFree.html.

2.2 Linking the L&S to LiLa

The LiLa KB includes about 200,000 canonical forms, each of which is described by a series of properties that record the part of speech (PoS), the full morphological description and the inflectional category. Also, the data property "written representation", defined in the ontology Ontolex (see Section 3.1), registers all the attested spellings of any lemma. Publishing a lexical resource as LLOD within LiLa means to both represent its information using the appropriate standards and vocabularies (Section 3.1) and to link the dictionary entries to the right form in LiLa by matching the lemmas used to index the records to the appropriate form in the KB.

In order to achieve the latter goal, firstly we had to normalise the spelling of the L&S dictionary lemmas by removing upper case initials and substituting *j* with *i* and *v* with *u* in order to mirror LiLa's conventions. Then, after mapping part-of-speech and inflectional information between resources, we extracted 31,142 1:1 matches, 2,998 1:N matches and 4,553 1:0 matches, on the basis of the tuple written representation - PoS. The latter group was subsequently matched only on the basis of graphical representation, at which point we obtained 946 1:1 matches and 50 1:N matches. Of the remaining 3,557 unmatched entries, 1,289 were successfully analysed by the morphological analyser Lemlat (Passarotti et al., 2017), leaving 2,239 definitely unmatched entries. After resolving multi-word spellings and graphical variants, the unmatched entries were all added to the LiLa Lemma Bank, while 1:N matches were manually disambiguated and matched to the relevant lemmas.

3 Modelling Lexical Entries

3.1 LiLa, Ontolex and lexicog

As said, the LiLa KB for Latin resources is built around a collection of canonical forms that can be used both as head words of dictionaries or as "targets" for the lemmatisation of corpora (Passarotti et al., 2020). These lemmas are modelled using the Ontolex ontology, a now *de facto* standard of the LLOD community (Cimiano et al., 2020; McCrae et al., 2017). In particular, lemmas in the LiLa KB are defined as forms of words that are linked (or are ready to be linked) to lexical entries via the property "canonical form" of the Ontolex ontology.[11]

Ontolex provides several classes and properties to describe the relationships that lexical entries have with, on the one hand, the grammatical forms attested in language and, on the other, the senses and the meanings of words. The core Ontolex module, however, imposes a series of restrictions that make its classes and properties ill-suited to represent the information in most standard dictionaries. The class Lexical Entry from the core Ontolex module, for instance, is inadequate to represent entries that license multiple syntactic interpretations, such as words that are registered in a dictionary as both adverb and conjunction. Subentries like the noun *verum* from the adjective *verus*, formed by a process of substantivisation from the word in the main entry, would also produce a mismatch between the dictionary and the lexical entry. Finally, the L&S, as most dictionaries, defines the senses of all but the most simple words by grouping them in sense clusters; those clusters are generally organized into hierarchies with multiple levels of nesting, from the most general to the most specific sense, a structure for which Ontolex has no suitable representation.

In order to overcome these issues, the Ontolex community has developed a specific extension of the ontology called the "OntoLex lexicography module" or `lexicog` (Bosque-Gil and Gracia, 2019).[12] The module is explicitly designed to capture the structural information expressed in a lexicographic resource and is primarily intended to support the conversion of lexicographic data that are not native to Ontolex. Retro-digitised dictionaries like the L&S are thus a perfect use case.

As said, `lexicog` focuses on the structural properties of dictionaries and does not attempt to convey any lexical, or indeed linguistic information, which are left to the classes and properties of Ontolex. The most important of these structural elements introduced in the vocabulary is that of the Lexicographic Entry. In `lexicog`, an entry is a container that represents a lexicographic article or record as it is arranged in the source (Bosque-Gil and Gracia, 2019). Thus, while a *lexical* entry (as defined in Ontolex) is an item in the lexicon of a given language, a *lexicographic* entry is a record in a linguistic resource that documents or discusses some properties of a given lexical item.

Lexicographic entries are a special subset of a larger class called Lexicographic Component. Apart from whole dictionary articles (the entries), components can be used to represent senses, sense groups or subentries (like the substantivised *verum*) within lexicographic entries.

It is important to stress once again that components represent only structural units; all linguistic information that is conveyed within these units must be expressed using Ontolex. The property `lexicog:describes` provides a link between the two dimensions, so that a lexicographic entry can be said to *describe* a lexical entry (as defined in Ontolex). In the same way, the lexicographic components that discuss a sense of a word or introduce a subentry, *describe* that specific lexical sense (as defined in Ontolex) or another lexical entry.

3.2 Lexicographic and Lexical Entries in the L&S

The LLOD version of the L&S linked to LiLa is now available online in the LiLa KB.[13] The entries can also be searched using LiLa's query interface and SPARQL endpoint.[14]

Figure 1 shows a visualisation of how the information from a sample entry, the adjective *hosticus* in the L&S dictionary, is represented in LiLa. In particular, the interplay between the linguistic and structural information is reflected in the complex relation between the lexical and lexicographic entries.

The L&S distinguishes two senses for the word: "belonging to an enemy, hostile" and "belonging to a stranger, foreign". Following the Ontolex approach, these meanings are represented by the two 'triangles' between the lexical entry (the light green node on the left), the concepts evoked by the word (gray-blue nodes), and the senses, labeled 0 ad 1, that mediate between them (greenish-yellow nodes).

The lexical entry is described by a *lexicographic entry*, identified by the id n21014 (inherited from the TEI XML file of the Perseus DL), while a specific lexicographic component describes each of the two senses (n21014_0 and n21014_1, respectively). What is particularly relevant is that the component n21014_0, which corresponds to the

[11] `http://www.w3.org/ns/lemon/ontolex#canonicalForm`.

[12] `https://www.w3.org/ns/lemon/lexicog#`.

[13] `http://lila-erc.eu/data/lexicalResources/LewisShort/Lexicon`.

[14] `https://lila-erc.eu/query/`, and `https://lila-erc.eu/sparql/`.

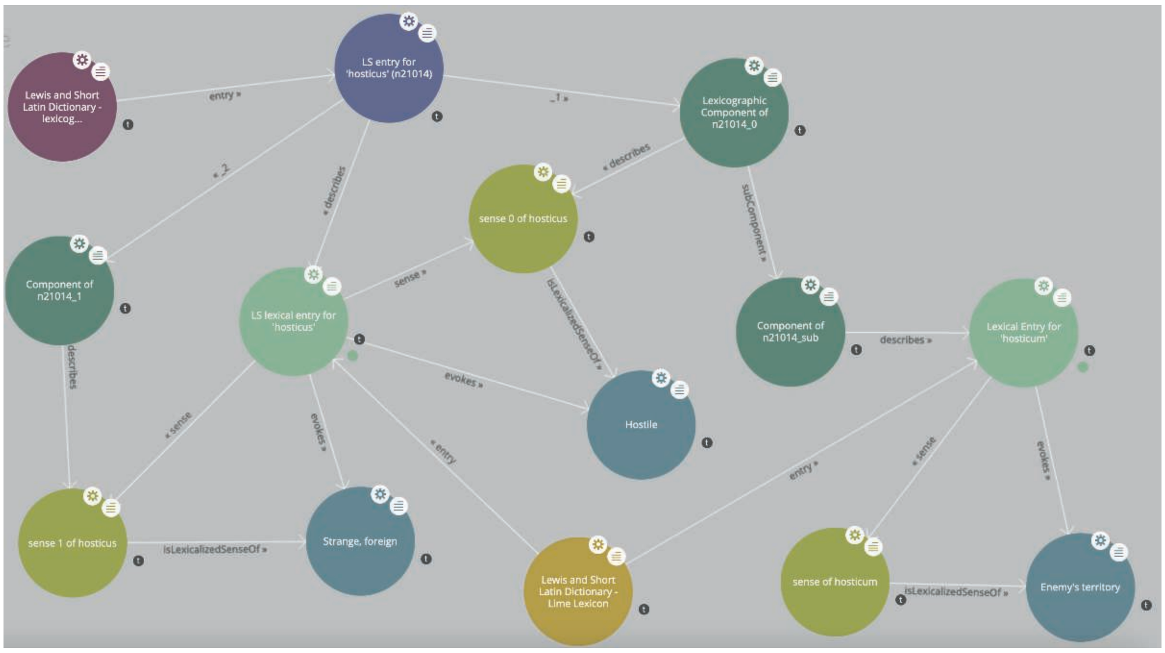


Figure 1: An entry in the LiLa's representation of the L&S.

sense "hostile", is linked to a sub-component that describes the lexical entry of the noun *hosticum*, a substantivised usage of the neuter adjective that means "the enemy's territory". That section of the entry that discusses the subentry "hosticum", which is itself a section of the paragraph dedicated to the first sense, is thus linked (via the "describes" property) to a different lexical entry.

4 Conclusions and Future Work

Perhaps even more than for any other modern language, a great number of lexical resources, either bi- or monolingual, is available for Latin, many of which have already been digitised and disseminated on the web. In this paper, we described a model of how this huge wealth of information can be published using the modern standards of the Semantic Web. The greatest advantage of this approach is that all the lexical resources published according to the same data model can be integrated in a wider network of linguistic information, along with the other digital resources that are connected to it. In the case of the L&S in LiLa, the Latin lexical entries of the bilingual dictionary can be queried together with the information about the same words provided by the other linguistic resources linked to the lemmas in the KB.

One example of the fruitful interactions between resources is the possibility to investigate the polysemy of words in relation to their derivation, as recorded in the Word Formation Latin resource, which is also linked to LiLa (Litta et al., 2020). The adjective *hosticus* of Figure 1, for instance, clearly inherits its two main senses ('hostile' and 'foreign') from the same polysemy of the noun *hostis* 'stranger' or 'enemy', from which it is derived. At the same time, while other resources in LiLa describe the senses of words, such as the Latin WordNet (Franzini et al., 2019; Mambrini et al., 2021), the complex relations between those senses (whether, for instance, one sense is interpreted as a specialised derivation from another) is generally available only in traditional lexical resources like the L&S.

The solutions we found to address the challenges raised by the representation of the L&S in LLOD will be reused when we will link further bilingual, as well as monolingual, dictionaries of Latin to the KB. Including such lexical resources in LiLa is an important achievement, as it makes it possible for the KB to interact with linguistic (meta)data for languages other than Latin. Undoubtedly, such an inter-linguistic (re)use of distributed resources is one of the objectives of the LLOD community, to which LiLa contributes by steadily providing it also with new (kinds of) linguistic resources represented in LLOD.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme – Grant Agreement No. 769994.

References

Richard Ashdowne, David R Howlett, and Ronald Edward Latham. 1975. *Dictionary of medieval Latin from British sources*. Oxford University Press.

David Bamman and Gregory Crane. 2011. The ancient greek and latin dependency treebanks. In *Language technology for cultural heritage*, pages 79–98. Springer.

Christian Bizer, Tom Heath, Kingsley Idehen, and Tim Berners-Lee. 2008. Linked data on the web (ldow2008). In *Proceedings of the 17th international conference on World Wide Web*, pages 1265–1266.

Julia Bosque-Gil and Jorge Gracia. 2019. The OntoLex lemon lexicography module. `https://ontolex.github.io/lexicog/`.

Flavio Massimiliano Cecchini, Timo Korkiakangas, and Marco Passarotti. 2020a. A new latin treebank for universal dependencies: Charters between ancient latin and romance languages. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 933–942.

Flavio Massimiliano Cecchini, Rachele Sprugnoli, Giovanni Moretti, and Marco Passarotti. 2020b. Udante: First steps towards the universal dependencies treebank of dante's latin works. In *CLiC-it*.

Philipp Cimiano, Christian Chiarcos, John P. McCrae, and Jorge Gracia. 2020. *Linguistic Linked Data: Representation, Generation and Applications*. Springer, Cham.

Irene De Felice, Giovanna Marotta, and Margherita Donati. 2015. Classes: A new digital resource for latin epigraphy. *IJCoL. Italian Journal of Computational Linguistics*, 1(1-1):125–136.

Joseph Denooz. 2007. Opera latina: le nouveau site internet du lasla. *Journal of Latin Linguistics*, 9(3):21–34.

Hanne Eckhoff, Kristin Bech, Gerlof Bouma, Kristine Eide, Dag Haug, Odd Einar Haugen, and Marius Jøhndal. 2018. The proiel treebank family: a standard for early attestations of indo-european languages. *Language Resources and Evaluation*, 52(1):29–65.

Wilhelm Ehlers. 1968. Der thesaurus linguae latinae. prinzipien und erfahrungen. *Antike und Abendland*, 14(1):172–184.

Egidio Forcellini and Jacobo Facciolati. 1871. *Lexicon totius latinitatis*, volume 3. Typis seminarii.

Greta Franzini, Andrea Peverelli, Paolo Ruffolo, Marco Passarotti, Helena Sanna, Edoardo Signoroni, Viviana Ventura, and Federica Zampedri. 2019. Nunc Est Aestimandum. Towards an evaluation of the Latin WordNet. In Raffaella Bernardi, Roberto Navigli, and Giovanni Semeraro, editors, *Sixth Italian Conference on Computational Linguistics (CLiC-it 2019)*, pages 1–8, Bari, Italy. CEUR-WS.org.

Peter GW Glare. 1968. *Oxford latin dictionary*. Clarendon Press, Oxford.

Nancy Ide and James Pustejovsky. 2010. What does interoperability mean, anyway? toward an operational definition of interoperability for language technology. In *Proceedings of the Second International Conference on Global Interoperability for Language Resources. Hong Kong, China*.

Steven E Jones. 2016. *Roberto Busa, SJ, and the emergence of humanities computing: the priest and the punched cards*. Routledge.

Neven Jovanović. 2012. Croala. enhancing a tei-encoded text collection. *Journal of the Text Encoding Initiative*, (2).

Ora Lassila, Ralph R. Swick, World Wide, and Web Consortium. 1998. Resource description framework (rdf) model and syntax specification.

Charlton T. Lewis and Charles Short. 1879. *A Latin Dictionary. Founded on Andrews' edition of Freund's Latin dictionary*. Clarendon Press, Oxford.

Henry Liddell, Robert Scott, and Henry Stuart Jones. 1940. *A Greek-English Lexicon*. Clarendon Press, Oxford, 9 edition.

Eleonora Litta, Marco Passarotti, and Francesco Mambrini. 2020. Derivations and Connections: Word Formation in the LiLa Knowledge Base of Linguistic Resources for Latin. *The Prague Bulletin Of Mathematical Linguistics*, 115:163–186.

Francesco Mambrini and Marco Passarotti. 2020. Representing etymology in the lila knowledge base of linguistic resources for latin. In *Proceedings of the 2020 Globalex Workshop on Linked Lexicography*, pages 20–28.

Francesco Mambrini, Marco Passarotti, Eleonora Litta, and Giovanni Moretti. 2021. Interlinking valency frames and wordnet synsets in the lila knowledge base of linguistic resources for latin. In *Further with Knowledge Graphs*, pages 16–28. IOS Press.

Massimo Manca, Linda Spinazzè, Paolo Mastandrea, Luigi Tessarolo, and Federico Boschetti. 2011. Musisque deoque: Text retrieval on critical editionse. *J. Lang. Technol. Comput. Linguistics*, 26(2):127–138.

John P. McCrae, Julia Bosque-Gil, Jorge Gracia, Paul Buitelaar, and Philipp Cimiano. 2017. The OntoLex-Lemon Model: development and applications. In *Proceedings of eLex 2017*, pages 587–597.

Barbara McGillivray and Alessandro Vatri. 2015. Computational valency lexica for latin and greek in use: a case study of syntactic ambiguity. *Journal of Latin Linguistics*, 14(1):101–126.

Stefano Minozzi. 2017. Latin wordnet, una rete di conoscenza semantica per il latino e alcune ipotesi di utilizzo nel campo dell'information retrieval. *Strumenti digitali e collaborativi per le Scienze dell'Antichità*, (14):123–134.

Marco Passarotti, Berta González Saavedra, and Christophe Onambele. 2016. Latin vallex. a treebank-based semantic valency lexicon for latin. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2599–2606.

Marco Passarotti, Marco Budassi, Eleonora Litta, and Paolo Ruffolo. 2017. The Lemlat 3.0 Package for Morphological Analysis of Latin. In Gerlof Bouma and Yvonne Adesam, editors, *Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language*, volume 133, pages 24–31, Gothenburg. Linköping University Electronic Press.

Marco Passarotti, Francesco Mambrini, Greta Franzini, Flavio Massimiliano Cecchini, Eleonora Litta, Giovanni Moretti, Paolo Ruffolo, and Rachele Sprugnoli. 2020. Interlinking through lemmas. the lexical collection of the lila knowledge base of linguistic resources for latin. *Studi e Saggi Linguistici*, 58(1):177–212.

Marco Passarotti. 2019. The project of the index thomisticus treebank. In *Digital Classical Philology*, pages 299–320. De Gruyter Saur.

Matteo Pellegrini, Eleonora Litta, Marco Passarotti, Francesco Mambrini, and Giovanni Moretti. 2021. The two approaches to word formation in the lila knowledge base of latin resources. In *Proceedings of the Third International Workshop on Resources and Tools for Derivational Morphology (DeriMo 2021)*, pages 101–109.

Johann Ramminger. 2008. *Neulateinische Wortliste. Ein Wörterbuch der Lateinischen von Petrarca bis 1700*. Thesaurus Linguae Latinae.

Jeffrey A Rydberg-Cox. 2002. Mining Data from an Electronic Greek Lexicon. *Classical Journal*, 98(2):183–188.

Rachele Sprugnoli, Francesco Mambrini, Giovanni Moretti, and Marco Passarotti. 2020. Towards the modeling of polarity in a latin knowledge base. In *WHiSe@ ESWC*, pages 59–70.

Paul Tombeur. 1998. *Thesaurus formarum totius Latinitatis: a Plauto usque ad saeculum XXum; TF.[2]. CETEDOC Index of Latin forms: database for the study of the vocabulary of the entire Latin world; base de données pour l'étude du vocabulaire de toute la latinité*. Brepols.

The ListTyp Database

Francesca Masini[1], **Simone Mattiola**[1], **Stefano Dei Rossi**[2]
1. Alma Mater Studiorum – University of Bologna, Italy
2. WebSoup, Italy
francesca.masini@unibo.it, simone.mattiola@unibo.it,
stefano@websoup.it

Abstract

English. The paper describes the aim and structure of a new freely accessible resource – *ListTyp: A typological database of listing patterns* – with a focus on methodological aspects, encoded information and search functions.

Italiano. *L'articolo descrive le finalità e la struttura di una nuova risorsa liberamente consultabile – ListTyp: A typological database of listing patterns – focalizzandosi su aspetti metodologici, informazioni codificate e funzioni di ricerca.*

1 Listing Patterns and Typology

Typological investigation is challenging in its own right, let alone when it tackles 'untraditional' categories, namely (newly-established) categories that are not part of the stock of customary, long-established concepts for linguistic description, hence not usually described in grammars, at all or as such. 'Lists' belong to this class.

Lists are traditionally associated with spoken language and interaction (see, among many others, Blanche-Benveniste (1990), Jefferson (1990), Selting (2007)). However, a broader approach has been proposed by Masini et al. (2018), who define 'lists' as syntagmatic concatenations of two or more units of the same type (potentially paradigmatically connected) that fill one and the same slot within the larger construction they are part of. This abstract definition embraces linguistic phenomena normally ascribed to different levels (morphology, syntax, discourse). 'Lists', or 'listing patterns' (LPs), thus encompass syntactic and discourse structures like coordination (e.g. *The system allows gas, electricity and water meters to be read* [British National Corpus]), reformulation (e.g. *They now had lifts, or rather elevators* [British National Corpus]) or repetition (e.g. *Some people are very very very touchy* [British National Corpus]), but also lexical and morphological phenomena like irreversible binomials (e.g. *alive and kicking*), (co)-compounding (e.g. Chuvash *sĕt-śu* lit. milk-butter 'dairy products', Wälchli (2005), p. 138) and full reduplication (e.g. Sundanese *hayan-hayan* lit. RED-want 'want very much', Moravcsik (1978), p. 321). Although these phenomena have their own specific properties (displaying different degrees of complexity, cohesion and conventionalization), lumping them together may unveil interesting (cross-linguistic) structural and functional tendencies and help bridging the gap between discourse and grammar.

Attempting a typological study of LPs is not trivial and raises methodological issues. Data are available for some widely described LPs (e.g. coordination, reduplication, co-compounding), but other types of LPs are far from simple to find in descriptive grammars, which usually (and understandably) focus on long-established categories in phonetics, morphology and syntax (leaving often aside, e.g., syntax beyond the clause and discourse phenomena). The same applies to typological databases. Hence, doing typology in the 'traditional' way turns out to be hard, and a new integrated methodology for carving out the required data is needed (Masini and Mattiola, 2019).

1.1 A Three-Level Methodology

The ListTyp database embodies this new methodology, which consists of three levels complementing each other (and running partially in parallel), encompassing both horizontal and vertical dimensions of investigation.

Firstly, a traditional large-scale examination of descriptive grammars is pivotal. For this first level

(**Level 1: horizontal**), a 'variety sample' (Miestamo et al., 2016) represents the best option.[1] This sample should be as large as possible (ideally 400-500 languages) to let the widest variety emerge. To this end, we have specifically created a sample of 424 languages (including isolate languages, pidgins/creoles and sign languages), following the Diversity Value technique with Ethnologue's 2018[2] genetic classification, which has proven to be the most reliable (Miestamo et al., 2016). Descriptive grammars for these languages were selected according to criteria such as: (i) exhaustivity (in terms of contents); (ii) searchability (digital edition); (iii) presence of (possibly glossed) texts; (iv) recentness. In order to facilitate the (time-consuming) process of data gathering, we subsequently created, from this larger sample, a smaller sample of 223 languages (with its own internal cohesion, based on the same 'variety' principles), which is what we are currently using to populate the database (cf. Mattiola (2020) for more details). Level 1 aims at achieving a preliminary survey of how languages work, but it merely scratches the surface: the general 'imperfections' of large-scale typology are made worse by the 'untraditional category' status of LPs, thus calling for other layers of investigation.

Secondly, a qualitative analysis of corpora and texts (e.g. texts at the end of descriptive grammars, free corpora, corpora made available by fieldworkers, etc.) is particularly useful to detect naturally occurring lists that are hard to be found in descriptive grammars used for Level 1. Needless to say, corpora of spoken language are especially useful for our current purposes. For this second level (**Level 2: intermediate**), the (convenience) sample is necessarily much smaller (ideally 20-30 languages). Level 2 maximizes the possibility to find discourse-level data (not necessarily described within the grammar) and allows to get over the problems of 'traditional' typology by verifying directly in a (albeit small) corpus data that the horizontal level did not bring out.

The third level, connected to the second, consists in a more quantitatively-oriented analysis of larger (possibly annotated) corpora of few (2-5) selected languages, which would provide enough data to draw some generalizations. Corpora might be either manually scrutinized (entirely or partially) or searched automatically through specific queries (depending on corpus annotation and size). The outputs of automatic searches are subsequently processed and checked manually. This level (**Level 3: vertical**) represents language-specific investigations that allow to study lists in much greater detail and to detect properties and constructions that more traditional methods might not be able to bring to light, as well as similarities between 'distant' languages.

The idea behind this three-level methodology is that combining data from different sources and extraction techniques not only enriches our database with new occurrences, but also contributes to unveil new patterns and to spot previously unexpected cross-linguistic correspondences. We believe that the very same methodology might be fruitfully applied to the typological investigation of other linguistic phenomena. At a more advanced stage of the project, we will also consider crowdsourcing as a collection technique, especially for underrepresented languages.

2 ListTyp Contents

ListTyp is an ongoing project: at present, the database is still only partially populated – counting **1685** examples of LPs from **156** languages – although its architecture is complete and freely available online: `https://listtyp.it/`.

The database is made of three main datasets (Dataset A, Dataset B, Dataset C) plus a supplement (Dataset D), each of which is partially independent, although they obviously concur to create the whole resource. Searches may be run on a single dataset or on the whole database.

Datasets A, B and C coincide with the three levels described in Subsection 1.1. They share the same architecture in terms of annotated properties and search criteria. However, they were gathered following (partially) different methodologies, which resulted in (partially) different sets of data, that are not directly comparable.

2.1 Dataset A

Dataset A is the result of Level 1 in our methodology, based on a large sample of typologically different languages. Hence, it represents the most 'typological' part of our database. Dataset A is being populated following the 223-language sample mentioned in Subsection 1.1 and currently con-

[1] A variety sample does not represent a balanced picture of the world's languages. Rather, it captures the broadest possible variation in order to maximize linguistic diversity.

[2] `https://www.ethnologue.com/`

tains 769 examples of LPs belonging to 152 languages. See the following example from Atayal: *musa' magaN qsinuw, ini' ga' piku' ru' ini' ga' bzwaq ru' ini' ga' yapit ga'* lit. ACT-go ACT-take animal NEG GA' squirrel and NEG GA' wild-pig and NEG GA' flying-squirrel GA' '(He) went to hunt animals: either squirrels, or wild pigs, or flying squirrels' (cf. Rau (1992), p. 188).

2.2 Dataset B

Dataset B is the result of Level 2 in our methodology, based on a much smaller sample of typologically different languages, which are analyzed through small-size (glossed) texts. The sample for this dataset is still undefined and is being built incrementally on the basis of availability. Languages to be included in Dataset B preferentially do not coincide with those included in Dataset A, but not necessarily. At present, Dataset B contains 72 examples of LPs from one language (Napoletano-Calabrese, Cilentan variety), extracted from a spoken corpus (e.g. *era tandu bella e tandu bella* '(She) was so nice and so nice').

2.3 Dataset C

Dataset C is the result of Level 3 in our methodology, based on few languages, which are however analyzed in a more thorough way using larger corpora. At present, Dataset C contains 661 occurrences from one language (Italian), taken from the spoken corpus LIP (De Mauro et al., 1993) (e.g. *è lui che organizza l'estorsioni le rapine i sequestri eccetera eccetera* 'He is the one who organizes extortion, robberies, kidnappings etcetera etcetera'). Further data from (spoken and written) Italian are being processed for inclusion in the database.

2.4 Dataset D: Supplement

The addition of a fourth dataset was necessary to document sparse examples collected in various ways by the ListTyp team and their students or other colleagues connected to the project. This supplement was therefore created without following any specific criterion, with the sole objective of enriching the resource. At present, Dataset D contains 183 lists (from written Italian, Russian and Spanish) connected to the COVID pandemic and manually gathered from Facebook (e.g. *No se van a controlar fiestas reuniones bares discotecas aforos* 'No control of parties, meetings, bars, discotheques, capacity will be carried out').

3 ListTyp Design

ListTyp is a web-based relational database containing a large number of parameters. Data, extracted with the different methods described in Subsection 1.1, were manually annotated by data collectors (whose contribution is acknowledged on the database website) under the supervision of the project directors.

3.1 Parameters

The main parameters, to be visualized on the 'Examples' webpage as a grid, include:

- *Language*: the name of the language according to Ethnologue (e.g. 'Tamasheq').
- *Source*: the type of source the example comes from (descriptive grammar, corpus, elicitation, web, social network, etc.).
- *Example*: the example as it appears in the original source (with no adjustments).
- *Glosses*: if the example was glossed in the original source, the original glosses are provided (with no adjustments, in most cases), otherwise they are added (in English) by the data collector.
- *Translation*: if the example was translated in the original source, the original translation is provided (with no adjustments)[3], otherwise it is added (in English) by the data collector.
- *Schema*: the abstract structural skeleton of the example (e.g. the schema for example *lifts, or rather elevators* would be 'X or Y').
- *Construction*: the grammatical phenomenon to which the example can be traced back, based on the commentary provided by the grammarian or the intuition of the fieldworker or data collector, despite the proliferation of terms this may entail. At present, ListTyp counts 13 values for this parameter[4], although the vast majority of examples are annotated as Coordination, Juxtaposition and Reduplication/repetition.

[3] Translations are mostly in English but also in other languages like French or Spanish.

[4] The values are: Alternative interrogatives; Co-compounding; Complex compounding; Compounding; Constrastive marker; Coordination; Coordination/list; Juxtaposition; List; Partial repetition list; Reduplication/repetition; Reformulation, Self-repair.

- *Function*: the function conveyed by the example based, again, on the commentary/translation provided by the grammarian or the intuition of the fieldworker or data collector. Here the proliferation of values is even more marked than for the 'Construction' parameter, as easily expected. At present, ListTyp counts 34 tags for this parameter[5], some of which are declared uncertain cases (like 'Plural / intensifying'), although there is a clear predominance of some functions like Additive and Alternative, but also Pluractional and Intensifying.[6]

By using the advanced search, other parameters are searchable, divided into three main groups of information: (i) Language info; (ii) Metadata; (iii) Formal and functional properties.

Information under **Language info** includes:

- *Iso Code 639 3*: the code for the representation of names of languages (Part 3).
- *Macro Area*: 'Africa', 'Australia', 'Australia & New Guinea', 'Eurasia', 'North America', 'South America'.
- *Family / Genus / Sub Classification*: following Ethnologue's genealogical classification.

Information under **Metadata** includes:

- *Reference*: the source (grammar, corpus, etc.) from which the example was taken.
- *Page*: the page or other reference – depending on the type of source – from which the example was taken.
- *Collector*: the person(s) responsible for (finding and/or uploading) the example.
- *Other Examples*: similar examples to be found in the same grammar (for the time being, only one example per type of structure is included in Dataset A).

[5]The values are: Additive; Additive / sequentiality; Adverbialization; Alternative; Alternative / approximating; Antipassive; Approximating; Attenuative; Categorizing; Clarification; Collective; Contrastive; Contrastive focus; Diminutive; Distributive; Emphasis; Endearment; Enumeration; Generalizing; Intensifying; Intensifying / pluractional; Nominalization; Non-prototypicality / plurality; Pluractional; Plural; Plural / intensifying; Politeness; Predicative; Reciprocal; Reformulation; Related variety; Self-repair; Skepticism; Stylistic effect; Word formation

[6]Both the 'Construction' and the 'Function' parameters and their values will be subject to reflection at a later stage of the project.

Information under **Formal and functional properties** (taken and adapted from Masini et al. 2018, to which we refer for details) includes:

- *Syndesis*: presence of connectives ('yes') (e.g. Kuot *U-rau, nəmo bun me-nəmu-a ga me-o* lit. 3mS-be.afraid COMPL APPR 3pS-kill-3mO and 3pS-eat.3sO 'He was afraid lest they kill and eat him', cf. Lindström (2002), p. 11) or absence of connectives ('no') (e.g. Lijili *Ziriji kè, móotòo kè, ńjìn kè* lit. train here-is, motor here-is, engine here-is 'There are trains and cars and engines'; cf. Stofberg (1978), p. 104).
- *Type Of Syndesis*: 'conjunctive' (cf. the Kuot example), 'disjunctive' (e.g. Yaul *Kawana mï mïnda o utam ama-p* lit. [name] 3SG banana or yam eat-PRF 'Kawana ate either a banana or a yam', Barlow (2018), p. 303) or 'adversative' (e.g. Madura *Hanina ngenom kopi tape banne teh* lit. Hanina AV.drink coffee but not tea 'Hanina drinks coffee but not tea', cf. Davies (2010), p. 339).
- *Prosodic Marking*: presence ('yes') or absence ('no') of (this field largely depends on the kind of source used and on the possibility to perform a prosodic analysis on the datum).
- *Type Of Prosodic Marking*: if present (open field).
- *Number Of Conjuncts*: the number of items that make up the LP example ('2', '3', '4', etc., up to very complex examples, like this from Italian, found in the LIP corpus (Dataset C): *RAIDUE o RAITRE o Canale cinque o Montecarlo Teleroma Gbr o Videomusic Retequattro chi piu' ne ha piu' ne vede* 'RAIDUE or RAITRE or Canale Cinque or Montecarlo Teleroma Gbr or Videomusic Retequattro whoever has more sees more').
- *Complexity Of Conjuncts*: 'Word', 'Phrase', 'Sentence'.
- *Category*: 'Nouns', 'Verbs', 'Adjectives', 'Adverbs', 'Numerals', etc. See for instance, in Gooniyandi, a case of reduplication of verbs (*doog* 'tap' > *doogdoog* 'tap repeatedly', cf. McGregor (1990), p. 83) vs. a case of reduplication of nouns (*barndanyi* 'old woman' > *barndanyibarndanyi* 'old women', cf. McGregor (1990), p. 237).

- *Presence Of Determiners*: 'yes' or 'no' (when the 'Category' is tagged as 'Nouns').

- *Dialogic*: 'yes' or 'no' (referring to the fact that lists may be dialogically co-constructed by speakers in interaction).

- *Interruption*: 'yes' or 'no' (referring to the fact that lists may be interrupted by, e.g., discourse markers or hesitations in interaction).

- *Type Of Interruption*: if present (open field).

- *Presence Of General Extender*: 'yes' or 'no' (general extenders being elements like *and stuff like that*, *and so on*, *etcetera* found at the end of a list, cf. Overstreet (2005)). See for instance Daga *ogi guep eragi kerip iravi* lit. banana loin/cloth mat betel/nut all 'banana, loin cloth, mat, and betel nut, all (of them)' (Murane (1974), p. 94) or Napoletano-Calabrese (Cilentan variety) *add'a ballà tutto 'u tribbunale // sègge // tavuli // tuttu còse!* lit. have.PRS.3SG COMPL dance.INF all DET court chairs tables all things 'It has to dance all the court: chairs, tables, all the things' (from Dataset B).

- *Type Of General Extender*: if present (open field).

- *Presence Of List Surroundings*: 'yes' or 'no' (list surroundings being elements connected to the LP that occur in its immediate context).

- *Type Of List Surroundings*: the values are 'projection component' or 'post-detailing component' (cf. Selting (2007)). In addition, the specific expression may be optionally added between square brackets. See e.g. this Italian example taken from the LIP corpus (Dataset C): *la seconda guerra mondiale e' [...] una guerra con armi piu' sofisticate bombe cioe' una guerra proprio di distruzione* 'World War II it's [...] a war with more sophisticated weapons bombs that is a war of destruction', where *cioe' una guerra proprio di distruzione* 'that is a war of destruction' is a post-detailing component.

- *Compositional*: 'yes' or 'no' (referring to the fact that lists may have different degrees of compositionality, a more or less literal/exhaustive interpretation, which we had to bring back to a binary value for simplicity). Reduplication examples like Lavukaleve *lafa* 'place' > *lafalafa* 'every place' (Terrill (2003), p. 36) or compounds like Kwewa, East *no'go-naaki* lit. girl-boy 'children' (Yarapea (2006), p. 169) are clear cases of non-compositional LPs, although non-literal, non-exhaustive lists are common in syntax too.

- *Natural Vs Accidental Coordination*: the possible values are 'natural' (marking that the conjuncts of the LP are lexico-semantically related, like in Havasupai-Walapai-Yavapai *had(a)-ch bos(a)-m day-k-yu* lit. dog-SUBJ cat-with 3=play=pl-ss-aux 'A dog and a cat are playing (together)'; cf. Watahomigie et al. (1982), p. 55) and 'accidental' (not lexico-semantically related, like in Gooniyandi *dawoonggoowaangginmiyi jaji maa-mi ngaaddi-mi* lit. you:two:like:it what meat-IND stone-IND 'Do you two want meat or money?', cf. McGregor (1990), p. 286), largely as intended by Wälchli (2005).

- *Semantic Relation Between Conjuncts*: the possible values are either the lexico-semantic relation between the conjuncts ('Synonyms', 'Co-hyponyms', 'Antonyms', etc.; plus 'Near-identical' / 'Identical') or the fact they are 'Frame-related' or 'Unrelated'.

Some fields may contain a double slash (//), which means that the field was deemed either irrelevant ('does not apply') or uncertain ('to be checked').

3.2 Search Options and Functions

Each of the parameters presented in Subsection 3.1 can be searched alone or in combination with other parameters. A specific set of filters can be saved and re-applied. The same holds for specific grid sorts. When performing a search, all valid hits appear in a tabular grid on the 'Examples' webpage.

3.3 Data Visualization

Data resulting from a query are visualized as text (relevant languages may be visualized on a map). The 'Examples' webpage shows the main parameters only, whereas the rest of the parameters are available through the 'Advanced search' interface. However, a function is available to personalize the main grid configuration in terms of page size, default filter criteria, default sort criteria, and order and display of grid columns.

Each single example in the database has three options of visualization (see the Appendix):

(i) as a line on the tabular grid, where each column corresponds to one of the main parameters (or the parameters customized and set by the user);

(ii) as a 'traditional' horizontal example with interlinear morphemic glosses (which shows up on request right below each line in the column grid);

(iii) as a separate full-page 'card' containing all the information available for that item, including main parameters, advanced search parameters, and localization map.

4 An Open Project

ListTyp is an ongoing project that welcomes collaborations for both data collection and analysis. We are currently processing data for completing Dataset A and enriching the other datasets. Updates will be published periodically. A full documentation will be available soon.

Acknowledgments

ListTyp is an outcome of *universaLIST – List constructions in typological and cognitive perspective*, a 3-year project (2017-2020) funded by the Department of Modern Languages, Literatures, and Cultures (LILEC) of the University of Bologna. The project is part of the research network *LIST – Listing in Natural Language* led by Francesca Masini and Caterina Mauri. The search interface and web design were built by WebSoup (Lucca, Italy): `https://www.websoup.it/`.

References

Russell Barlow. 2018. *A grammar of Ulwa*. University of Hawai'i at Mānoa Doctoral Dissertation, Mānoa.

Claire Blanche-Benveniste. 1990. Un modèle d'analyse syntaxique "en grilles" pour les productions orales. *Anuario de Psicología*, 47:11–28.

William D. Davies. 2010. *A grammar of Madurese*. Mouton de Gruyter, Berlin/New York.

Tullio De Mauro, Federico Mancini, Massimo Vedovelli, and Miriam Voghera. 1993. *Lessico di frequenza dell'italiano parlato*. Etaslibri, Milano.

Gail Jefferson. 1990. List-construction as a task and resource. In George Psathas, editor, *Interactional competence*, pages 63–92. Irvington Publishers, New York.

Eva Lindström. 2002. *Topics in the Grammar of Kuot*. Stockholm University Doctoral Dissertation, Stockholm.

Francesca Masini and Simone Mattiola. 2019. Come fare tipologia con categorie non tradizionali? In Chiara Gianollo and Caterina Mauri, editors, *CLUB Working Papers in Linguistics 3*, pages 282–294. CLUB – Circolo Linguistico dell'Università di Bologna, Bologna.

Francesca Masini, Caterina Mauri, and Paola Pietrandrea. 2018. List constructions: Towards a unified account. *Italian Journal of Linguistics*, 30(1):49–94.

Simone Mattiola. 2020. Two language samples for maximizing linguistic variety. Alma Mater Studiorum - Università di Bologna, Bologna.

William McGregor. 1990. *A Functional Grammar of Gooniyandi*. John Benjamins, Amsterdam/Philadelphia.

Matti Miestamo, Dik Bakker, and Antti Arppe. 2016. Sampling for variety. *Linguistic Typology*, 20(2):233–296.

Edith Moravcsik. 1978. Reduplicative constructions. In Joseph Greenberg, editor, *Universals of human language*, volume 3: Word Structure, pages 297–334. Stanford University Press, Stanford.

Elizabeth Murane. 1974. *Daga grammar: From morpheme to discourse*. The Summer Institute of Linguistics and the University of Texas at Arlington, Norman.

Maryann Overstreet. 2005. And stuff und so: Investigating pragmatic expressions in English and German. *Journal of Pragmatics*, 37(11):1845–1864.

Der-Hwa Victoria Rau. 1992. *A Grammar of Atayal*. UMI [Cornell University Doctoral Dissertation], Ann Arbor.

Margret Selting. 2007. Lists as embedded structures and the prosody of list construction as an interactional resource. *Journal of Pragmatics*, 39(3):483–526.

Yvonne F. Stofberg. 1978. *Migili grammar*. The Summer Institute of Linguistics, Dallas.

Angela Terrill. 2003. *A Grammar of Lavukaleve*. Mouton de Gruyter, Berlin/New York.

Lucille J. Watahomigie, Jorigine Bender, and Akira Y. Yamamoto. 1982. *Hualapai reference grammar*. American Indian Studies Center, UCLA, Los Angeles.

Bernhard Wälchli. 2005. *Co-compounds and natural coordination*. Oxford University Press, New York.

Apoi Mason Yarapea. 2006. *Morphosyntax of Kewapi*. Australian National University Doctoral Dissertation, Berlin.

Appendix: Visualizations for Example 269

Tabular grid

Language	Source	Example	Glosses	Translation	Schema	Construction	Function		
Yeri	Grammar	yot-u-ø h-o **mɨneigi wia-i wia-i o mɨneigi ŋa-i o wona ŋa-n o**	DEM-MDIST-SG.F 1PL-stay.R **time.period two-F two-F or time.period one-PL or moon one-SG.M or**	'There we stayed for **four weeks or one week or one month.**'	X or Y or Z or	Coordination	Approximating		

Horizontal

yot-u-ø h-o ***mɨneigi wia-i wia-i o mɨneigi ŋa-i o wona ŋa-n o***
DEM-MDIST-SG.F 1PL-stay.R **time.period two-F two-F or time.period one-PL or moon one-SG.M or**
'There we stayed for **four weeks or one week or one month.**'

Full-page 'card'

Available at:
`https://listtyp.it/row/view?id=269`

LIST TYP A typological database of listing patterns

Home Project Examples Who we are & contacts Related works

Example: #269

Language	Yeri
Source	Grammar
Example	yot-u-ø h-o **mɨneigi wia-i wia-i o mɨneigi ŋa-i o wona ŋa-n o**
Glosses	DEM-MDIST-SG.F 1PL-stay.R **time.period two-F two-F or time.period one-PL or moon one-SG.M or**
Translation	'There we stayed for **four weeks or one week or one month.**'
Schema	X or Y or Z or
Construction	Coordination
Function	Approximating

Metadata

Reference	Jennifer Wilson. 2017. A grammar of Yeri: A Torricelli language of Papua New Guinea. (Doctoral dissertation, State University of New York at Buffalo; xxviii+805pp.)
Page	337
Collector	Simone Mattiola
Other Examples	
Status	

Language info

Iso Code 639 3	yev
Macro Area	Australia & New Guinea
Family	Torricelli
Genus	Wapei-Palei
Sub Classification	Wapei

Formal and functional properties

Syndesis	Yes
Type Of Syndesis	Disjunctive
Prosodic Marking	No
Type Of Prosodic Marking	//
Number Of Conjuncts	3
Complexity Of Conjuncts	Phrase
Category	NPs
Presence Of Determiners	//
Dialogic	No
Interruption	No
Type Of Interruption	//
Presence Of General Extender	No
Type Of General Extender	//
Presence Of List Surroundings	No
Type Of List Surroundings	//
Compositional	No
Natural Vs Accidental Coordination	Natural
Semantic Relation Between Conjuncts	Co-hyponyms

Probing Tasks Under Pressure

Alessio Miaschi[1,2], **Chiara Alzetta**[1], **Dominique Brunato**[1], **Felice Dell'Orletta**[1], **Giulia Venturi**[1]
[1]Istituto di Linguistica Computazionale "Antonio Zampolli", Pisa
ItaliaNLP Lab – *www.italianlp.it*
[2]Department of Computer Science, University of Pisa
alessio.miaschi@phd.unipi.it, name.surname@ilc.cnr.it

Abstract

Probing tasks are frequently used to evaluate whether the representations of Neural Language Models (NLMs) encode linguistic information. However, it is still questioned if probing classification tasks really enable such investigation or they simply hint for surface patterns in the data. We present a method to investigate this question by comparing the accuracies of a set of probing tasks on gold and automatically generated control datasets. Our results suggest that probing tasks can be used as reliable diagnostic methods to investigate the linguistic information encoded in NLMs representations.

1 Introduction

In recent years we saw the raise of a consistent body of work dealing with the use of probing tasks to test the linguistic competence learned by Neural Language Models (NLMs) (Conneau et al., 2018; Warstadt et al., 2019; Hewitt and Liang, 2019; Miaschi et al., 2020). The idea behind the probing paradigm is actually quite simple: using a diagnostic classifier, the *probing model* or *probe*, that takes the output representations of a NLM as input to perform a *probing task*, e.g. predict a given language property. If the probing model will predict the property correctly, then we can assume that the representations somehow encode that property. Studies relying on this method reported that NLMs representations do encode several properties related to morphological, syntactic and semantic information.

Despite the amount of work, there are still several open questions concerning their use (Belinkov, 2021): which probing model should we use for assessing the linguistic competence of a NLM? Are probes the most effective strategy to achieve such goal? These questions fostered two complementary lines of research. The first one is devoted to modifying the architecture of the current probing models; the other one is focused on evaluating the effectiveness of probing models. Both are still not well investigated issues, although their importance for advancing the research on the evaluation of NLMs linguistic competences has been widely recognized.

Among the first line of research, dealing with the design of probing classifiers, several works investigate which model should be used as probe and which metric should be employed to measure their performance. With this respect, it is still questioned if one should rely on simple models (Hewitt and Manning, 2019; Liu et al., 2019; Hall Maudslay et al., 2020) or complex ones (Pimentel et al., 2020; Voita and Titov, 2020) in terms of model parametrization. Specifically, Voita and Titov (2020) suggest to design alternative probes using a novel information-theoretic approach which balances the probe inner complexity with its task performance.

Concerning works facing the issue of investigating the effectiveness of the probing paradigm, Hewitt and Liang (2019) observe that probing tasks might conceal the information about the NLM representation behind the ability of the probe to learn surface patterns in the data. To test this idea, they introduced *control tasks*, a set of tasks that associate word types with random outputs that can be solved by simply learning regularities. Along the same line, Ravichander et al. (2021) test probing tasks by creating control datasets where a property is always reported in a dataset with the same value, thus it is not discriminative for testing the information contained in the representations. Their experiments highlight that the probe may learn a property also incidentally, thus casting doubts on the effectiveness of probing tasks.

The scenario defined by the latter two works is the one we deal with in this paper. Specifically, we introduce a new approach to put increasingly under pressure the effectiveness of a suite of probing tasks to test the linguistic knowledge implicitly encoded by BERT (Devlin et al., 2019), one of the most prominent NLMs. To achieve this goal, we set up a number of experiments (see Section 2) aimed at comparing the performance of a regression model trained with BERT representations to predict the values of a set of linguistic properties extracted from the Italian Universal Dependency Treebank (Zeman et al., 2020) and from a suite of *control datasets* we specifically built for the purpose of this study. We define a control dataset as a set of linguistic features whose values were automatically altered in order to be increasingly different from the values in the treebank, referred to as *gold* values. Our underlying hypothesis is that if the predictions of the increasingly altered values progressively diverge from the predictions of the gold values, this possibly suggests that the corresponding probing tasks are effective strategies to test the linguistic knowledge embedded in BERT representation We will discuss the results of our experiments in light of this hypothesis in Section 3. In Section 4 we will draw the conclusions.

Note that this is one of the few studies focused on non-English NLMs. In fact, with the exception of (de Vries et al., 2020; Miaschi et al., 2021; Guarasci et al., 2021), the majority of research related to interpretability issues is focused on English or, at most, multilingual models.

Contributions To the best of our knowledge this is the first paper that (i) introduces a methodology to test the reliability of probing tasks by building control tasks at increasing level of complexity, (ii) puts under pressure the probing approach considering the Italian language.

2 Methodology

Our methodology seeks to investigate the effectiveness of probing tasks for evaluating the linguistic competences encoded in NLM representations. To this aim, we trained a probing model (described in Section 2.1) using BERT sentence representations and then tested its performance when predicting the values of a set of linguistic features (see Section 2.3) in multiple scenarios. In one scenario, the model shall predict gold values, thus corresponding to the real values of the features in the corpus. In the other scenarios, we automatically altered the feature values at different control levels each corresponding to increasing degrees of pressure for the probing model, as discussed in Section 2.4.

Our methodology will allow us to test whether the probing model really encodes linguistic competences or simply learns regularities in the task and data distributions by checking the results obtained in the different scenarios. If the predictions of the probing model will be more similar to the gold values than to the automatically altered ones, then we might assume that the information captured by the probed feature is encoded in the representations.

2.1 Model

Our model is a pre-trained Italian BERT. Specifically, we used the base cased BERT developed by the MDZ Digital Library Team, available trough the Huggingface's *Transformers* library (Wolf et al., 2020)[1]. The model was trained using Wikipedia and the OPUS corpus (Tiedemann and Nygaard, 2004). For the sentence-level representations, we leveraged the activation of the first input token *[CLS]*. The probing model is a linear Support Vector Regression model (LinearSVR).

2.2 Data

Our experiments are carried out on the Italian Universal Dependencies Treebank (IUDT), version 2.5 (Zeman et al., 2020), containing a total of 35,480 sentences. Due to the IUDT high variability in terms of sentence length[2], we focused on a sub-set of sentences with a ± 10 tokens variation with respect to the median sentence length (i.e. 20 tokens). As a result, we selected 21,991 sentences whose length ranges between 10 and 30 tokens. This way our dataset is balanced, viz., the amount of sentences with exact same length considered for the experiments is comparable. Specifically, our dataset accounts for around 1,000 sentences for each reported value of sentence length, which makes the results of our analyses reliable and comparable.

[1] https://huggingface.co/dbmdz/bert-base-italian-xxl-cased

[2] IUDT contains sentences ranging from 1 to 308 token long.

Morphosyntactic information
Distibution of UD POS
Lexical density
Inflectional morphology
Distribution of lexical verbs and auxiliaries for inflectional categories (tense, mood, person, number)
Verbal Predicate Structure
Distribution of verbal heads and verbal roots
Average verb arity and distribution of verbs by arity
Global and Local Parsed Tree Structures
Depth of the whole syntactic tree
Average length of dependency links and of the longest link
Average length of prepositional chains and distribution by depth
Average clause length
Relative order of elements
Distribution of subjects and objects in post- and pre-verbal position
Syntactic Relations
Distribution of dependency relations
Use of Subordination
Distribution of subordinate and principal clauses
Average length of subordination chains and distribution by depth
Distribution of subordinates in post- and pre-principal clause position

Table 1: Linguistic features probed in the experiments.

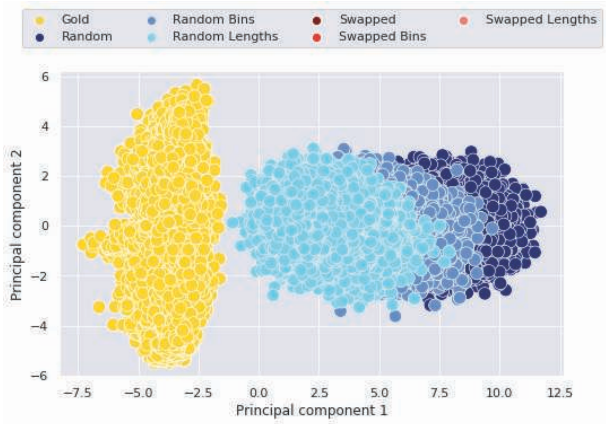


Figure 1: 2-dimensional PCA projection of the feature values in the gold and control datasets. All *Swapped* datasets overlap with the *Gold* one.

2.3 Linguistic Features

The probing tasks we defined consist in predicting the value of multiple linguistic features, each corresponding to a specific property of sentence structure. The set includes 77 linguistic features and it is based on the ones described in Brunato et al. (2020) modeling 7 main aspects of the structure of a sentence, which are reported in Table 1. They range from morpho-syntactic and inflectional properties, to more complex aspects of sentence structure (e.g. the depth of the whole syntactic tree), to features referring to the structure of specific sub-trees, such as the order of subjects and objects with respect to the verb, to the use of subordination.

We chose to rely on these features for two main reasons. Firstly, they have been shown to be highly predictive when leveraged by traditional learning models on a variety of classification problems where the linguistic information plays a fundamental role. In addition, they are multilingual as they are based on the Universal Dependency formalism for sentence representation (Nivre, 2015). In fact, they have been successfully used to profile the knowledge encoded in the language representations of contextual NLMs for both the Italian (Miaschi et al., 2021) and English language (Miaschi et al., 2020).

In this study, the values of each feature acquired from IUDT represent the *gold dataset* and they have been automatically altered in order to generate additional *control datasets*.

2.4 Control Datasets

We created two main types of control datasets, obtained by automatically altering gold feature values. The first main type (hereafter referred to as *Swapped*) is built by shuffling the original values of each feature across sentences; while the second type (*Random*) contains values randomly generated within the maximum and the minimum value that each feature shows in the whole gold dataset. To clarify, consider the following example involving the feature `average link length`, which captures the average linear distance between dependents and their syntactic head within a sentence. In the *Swapped* variant we simply swap the feature values, thus a sentence which originally showed an `average link length` of, e.g., 2.86 could be changed to 8.83. Note that both are real values extracted from our dataset. When building the *Random* variant, all sentences considered for the study show a feature value randomly generated between 1.33 and 9.78, which are the reported minimum and maximum `average link length` values in the dataset, respectively associated to sentences with length 11 and 21.

Since the values of the considered features are strongly related to the length of the sentence, for each type of control dataset we built two sub-types of datasets. In a first sub-type (*Bins*), we grouped sentences falling into the same predefined range of sentence lengths (i.e., 10-15, 15-20, 20-25 and 25-30 tokens). In a second sub-type (*Lengths*), we included groups of sentences having exactly the same length. This motivates the choice of

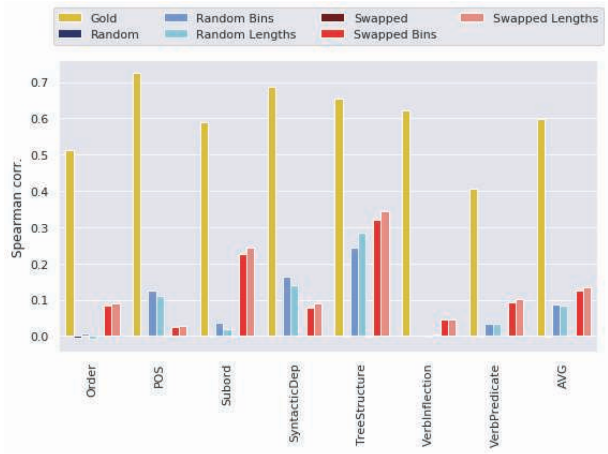


Figure 2: Average probing scores (as Spearman correlation) obtained by the LinearSVR model when predicting *gold* and *control* linguistic features. Results are reported for each feature group and on average ('AVG' column).

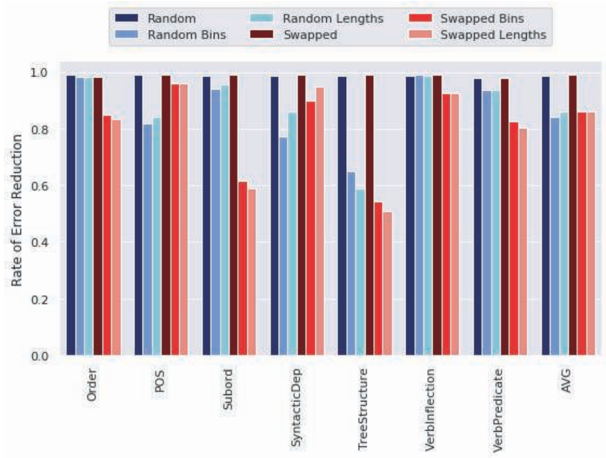


Figure 3: Error reduction rates reporting the difference between the probing scores obtained on the Gold dataset and each control dataset. Result are reported for each feature group and on average ('AVG' column).

sentences whose length ranges in an interval for which we have a reliable amount of instances (as introduced in Section 2.2).

Note that the different data altering strategies are conceived to represent increasingly challenging testbeds to assess the effectiveness of our probing tasks. The *Swapped* control datasets are the most challenging ones as the swapped feature values might be quite similar to the gold ones, thus possibly predicted with an high accuracy by the probing model. Such intuition is confirmed by the results of the 2-dimensional Principal Component Analysis (PCA) reported in Figure 1[3]. As we can see, all the data points representing the feature values contained in the *Swapped* datasets fully overlap with the gold ones, thus confirming their similarity. On the contrary, randomly generated values are progressively more distant being less plausible, even if the constraints of sentence length yield values that are closer to the gold ones.

3 Results

For both gold and control datasets, probing scores are computed as a Spearman correlation between the feature values predicted by the probing model and the values contained in each dataset. Such correlation values are computed by averaging the NLM's layer–wise scores as, for all datasets, we observed small differences between the scores obtained across the 12 layers. We experimentally verified that these differences were not significant by computing the slope of a linear regression line between BERT layers and the scores of the gold dataset, obtaining -0.0017 as mean value considering all features. Our intuition is that the small range of lengths of the sentences here considered may have yielded such insignificant variation across layers, which on the contrary Miaschi et al. (2021) showed to be significant on the whole set of IUDT sentences. Namely, being highly related to the length of the sentence, the feature values have little variations. However, a more in-depth investigation of the underlying reasons of this outcome is one of the future directions of this work.

Figure 2 shows the scores obtained on the gold and the 6 control datasets, both for the 7 macro-groups of linguistic features and on average (*AVG*). Additionally, in order to properly appreciate the differences between the results obtained on the *gold* and control datasets, in Figure 3 we report the error reduction rate for each control dataset computed as the difference between the scores obtained when predicting gold and altered features.

General Results. We can observe that on average the highest probing scores are obtained on the gold dataset and that, accordingly, there is a great difference (i.e. almost 1.0, see Figure 3) between

[3]PCA is a classical data analysis method that reduces the dimensionality of the data while retaining most of the variation in the data set by identifying n principal components, along which the variation of the data is maximal (Jolliffe and Cadima, 2016).

the accuracy of the probing model when predicting the authentic and altered feature values. This seems suggesting that the model is able to recognize that the feature values contained in the control datasets have been altered, even when they are not fully random but plausible, i.e. in the *Swapped* datasets. As a consequence, we can hypothesize that the model is relying on some implicit linguistic knowledge when it predicts the authentic feature values, rather than learning some regularities possibly found in the dataset.

However, if we take a closer look at the scores obtained for the *Random* and *Swapped* datasets when we constrain the length of the sentences, we can observe that the accuracy in predicting the feature values contained in the *Swapped* datasets is sightly higher than in the *Random* ones (see 'AVG' column in Figure 2). This is in line with our starting hypothesis and shows that feature values artificially created simply by shuffling gold ones across sentences of the same lengths (or of the same range of lengths) are more similar to the gold values and thus are predicted with higher accuracy than randomly altered values. Nevertheless, their error rate, namely the difference from the accuracy of gold predictions, is still quite high, i.e. about 0.80 (see the 'AVG' column, Figure 3).

Linguistic Features Analysis. Also when we focus on the results obtained with respect to the 7 macro-groups of linguistic features, we can observe that the probing model is more accurate in the prediction of the gold values. Again, the scores on the control datasets are slightly higher when we constrain the values with respect to sentence length, since we narrow the range of possible values. In particular, we see that the feature values related to the sentence tree structure are those predicted most closely to the gold ones (see column 'TreeStructure', Figure 3). Note that these sentence properties are the most sensitive to the sentence length, that BERT encodes with a very high accuracy. This may suggest that in the resolution of these tasks the probing model is possibly relying on some regularities related to sentence length.

Similar observations hold for the results achieved in the resolution of the probing tasks related to the use of subordination, which heavily depends on sentence length. Interestingly, we can note that the values of all the other groups of features contained in the control datasets are predicted by the probing model with a very low accuracy, possibly making the results not significant.

Dataset	**Spearman correlation**
Random	0.08
Random Bins	0.46 *
Random Lengths	0.33 *
Swapped	-0.15
Swapped Bins	0.05
Swapped Lengths	0.06

Table 2: Spearman correlations between the rankings of features obtained with the *Gold* dataset and the 6 control datasets. Statistically significant correlations are marked with * (p-value < 0.05).

Gold	**Random Bins**	**Swapped Lengths**
dep_dist_root	dep_dist_root	dep_dist_root
dep_dist_punct	avg_max_links_len	avg_max_links_len
upos_dist_PUNCT	max_links_len	max_links_len
xpos_dist_FS	xpos_dist_FB	avg_max_depth
upos_dist_ADP	avg_token_per_clause	verbal_head_per_sent
dep_dist_det	xpos_dist_FS	xpos_dist_FS
upos_dist_PROPN	n_prep_chains	avg_links_len
upos_dist_DET	avg_max_depth	subord_prop_dist
xpos_dist_RD	verbal_head_per_sent	avg_subord_chain_len
dep_dist_case	xpos_dist_RI	n_prep_chains
verbal_head_per_sent	dep_dist_cop	subord_post
xpos_dist_FF	xpos_dist_PC	subord_dist_1
xpos_dist_SP	dep_dist_conj	avg_prep_chain_len
xpos_dist_E	xpos_dist_B	obj_post
upos_dist_NOUN	xpos_dist_VA	avg_verb_edges

Table 3: 15 top-ranked *Gold* and control features (*Random Bins* and *Swapped Lengths*) predicted by BERT sentence-level representations.

Features Correlations. Once we showed that the probing tasks accuracy is very different if the feature values are authentic or altered, in this section we compare the ranking of linguistic features ordered by decreasing prediction accuracy in the gold and control scenarios. As we can see in Table 2, which reports the Spearman correlations between the rankings, the *control rankings* are almost not related to the gold one and the existing correlations in most cases are not even statistically significant. The only exceptions are represented by the rankings of values that were randomly generated with sentence length constraints, which have a weak and moderate correlation. Note that however, as shown before, the probing scores are very low.

A more qualitative feature ranking analysis can be carried out by inspecting Table 3 where we report the first 15 top-ranked features predicted in the gold and in the two most highly correlated *Swapped* and *Random* datasets. As we can see, the *gold ranking* diverges from the rankings of the altered values with respect to the majority of top-ranked features. The most visible exception is represented by the distribution of syntactic root (*dep_dist_root*) that the probing model always predicts with the highest accuracy. The result is quite expected since this feature can be seen as a proxy of the length of the sentence, a linguistic property properly encoded by BERT. Similarly, other two features influenced by sentence length appear, as expected, on the top positions of all rankings, namely the distribution of the sentence boundary punctuation (*xpos_dist_FS*) and of verbal heads (*verbal_head_per_sent*).

4 Discussion and Conclusion

In this paper we described a methodology to test the effectiveness of a suite of probing tasks for evaluating the linguistic competence encoded by NLMs. To this aim, we analysed the performance of a probing model trained with BERT representations to predict the authentic and automatically altered values of a set of linguistic features derived from IUDT. We observed general higher performance in the prediction of authentic values, thus suggesting that the probing model relies on linguistic competences to predict linguistic properties. However, when we constrained automatically altered values with respect to sentence length, the model tends to learn surface patterns in the data.

As a general remark, it should be pointed out that our analyses dealt only with sentences showing a standard length (i.e., between 10 and 30 tokens per sentence). This choice, if on the one hand made our results more directly comparable across bins of sentences sharing the same length, on the other hand excluded from the analyses the shortest and the longest sentences of IUDT. Our future work will be devoted to replicate the probing task experiments described in this paper also on control datasets comprising sentences whose length is outside of the range considered here. To this aim, we performed preliminary analyses to test the scores of probing tasks on gold IUDT sentences that are less than 10-token and more than 30-token long. Interestingly, we noticed that the probing model is less accurate when predicting the linguistic features extracted from the group of IUDT short sentences. Specifically, the average Spearman correlation obtained on such group is 0.47, while probing scores on longer sentences (+30-token long) and on those used in our experiments achieved an average correlation of 0.56 and 0.66 respectively. Starting from this preliminary finding, a possible future investigation could focus on whether using longer or shorter sentences would also have an effect on the probing scores obtained with the control datasets.

In future work we also plan to investigate which features are more diagnostic of the linguistic competence encoded by a NLM and which ones, on the contrary, are more influenced by confounders, such as sentence length.

References

Yonatan Belinkov. 2021. Probing Classifiers: Promises, Shortcomings, and Advances. *Computational Linguistics*, pages 1–12, 10.

Dominique Brunato, Andrea Cimino, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2020. Profiling-UD: a tool for linguistic profiling of texts. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 7147–7153, Marseille, France, May. European Language Resources Association.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia, July. Association for Computational Linguistics.

Wietse de Vries, Andreas van Cranenburgh, and Malvina Nissim. 2020. What's so special about BERT's layers? a closer look at the NLP pipeline in monolingual and multilingual models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4339–4350, Online, November. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Raffaele Guarasci, Stefano Silvestri, Giuseppe De Pietro, Hamido Fujita, and Massimo Esposito. 2021. Assessing BERT's ability to learn Italian syntax: a study on null-subject and agreement phenomena. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–15.

Rowan Hall Maudslay, Josef Valvoda, Tiago Pimentel, Adina Williams, and Ryan Cotterell. 2020. A tale of a probe and a parser. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7389–7395, Online, July. Association for Computational Linguistics.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743.

John Hewitt and Christopher D Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138.

Ian T Jolliffe and Jorge Cadima. 2016. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Alessio Miaschi, Dominique Brunato, Felice Dell'Orletta, and Giulia Venturi. 2020. Linguistic profiling of a neural language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 745–756, Barcelona, Spain (Online), December. International Committee on Computational Linguistics.

Alessio Miaschi, Gabriele Sarti, Dominique Brunato, Felice Dell'Orletta, and Giulia Venturi. 2021. Italian transformers under the linguistic lens. In Johanna Monti, Felice Dell'Orletta, and Fabio Tamburini, editors, *Proceedings of the Seventh Italian Conference on Computational Linguistics (CLiC-it 2020)*, Online, March. CEUR Workshop Proceedings (CEUR-WS.org).

Joakim Nivre. 2015. Towards a universal grammar for natural language processing. In *Proceedings of The 16th Annual Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, pages 3–16. Springer.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4609–4622.

Abhilasha Ravichander, Yonatan Belinkov, and Eduard Hovy. 2021. Probing the probing paradigm: Does probing accuracy entail task relevance? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3363–3377, Online, April. Association for Computational Linguistics.

Jörg Tiedemann and Lars Nygaard. 2004. The OPUS corpus - parallel and free: `http://logos.uio.no/opus`. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, May. European Language Resources Association (ELRA).

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 183–196.

Alex Warstadt, Yu Cao, Ioana Grosu, Wei Peng, Hagen Blix, Yining Nie, Anna Alsop, Shikha Bordia, Haokun Liu, Alicia Parrish, Sheng-Fu Wang, Jason Phang, Anhad Mohananey, Phu Mon Htut, Paloma Jeretic, and Samuel R. Bowman. 2019. Investigating BERT's knowledge of language: Five analysis methods with NPIs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2877–2887, Hong Kong, China, November. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October. Association for Computational Linguistics.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Noëmi Aepli, Željko Agic, Lars Ahrenberg, et al. 2020. Universal dependencies 2.5. *LINDAT/CLARIAHCZ digital library at the Institute of Formal and Applied Linguistics (UFAL), Faculty of Mathematics and Physics, Charles University. url: http://hdl. handle. net/11234/1-3226*.

Frame Semantics for Social NLP in Italian: Analyzing Responsibility Framing in Femicide News Reports

Gosse Minnema[1], **Sara Gemelli**[2], **Chiara Zanchi**[2], **Viviana Patti**[3], **Tommaso Caselli**[1], **Malvina Nissim**[1]
1. University of Groningen, The Netherlands
2. University of Pavia, Italy
3. University of Turin, Italy
{g.f.minnema,t.caselli,m.nissim}@rug.nl, chiara.zanchi01@unipv.it, sara.gemelli01@universitadipavia.it, patti@di.unito.it

Abstract

We propose using a FrameNet-based approach for analyzing how socially relevant events are framed in media discourses. Taking femicides as an example, we perform a preliminary investigation on a large dataset of news reports and event data covering recent femicides in Italy. First, we revisit the EVALITA 2011 shared task on Italian frame labeling, and test a recent multilingual frame semantic parser against this benchmark. Then, we experiment with specializing this model for Italian and perform a human evaluation to test our model's real-world applicability. We show how FrameNet-based analyses can help to identify linguistic constructions that background the agentivity and responsibility of femicide perpetrators in Italian news.

1 Introduction

Frame semantics (Fillmore, 1985; Fillmore, 2006) is a theory of natural language understanding with a focus on word meanings (*lexical units*) and semantic roles (*frame elements*). The associated FrameNet project (Baker et al., 2003) has resulted in an extensive lexicon and annotated corpus implementing this theory. In the Italian computational linguistics community, there has also been considerable work on frame semantics, mostly focused on creating FrameNet resources (Tonelli and Pianta, 2008; Tonelli et al., 2009; Lenci et al., 2010; Basili et al., 2017; Brambilla et al., 2020). However the practical usability of frame semantics for Italian is still largely unexplored. First of all, on automatic *frame semantic parsing* (FSP) (Gildea and Jurafsky, 2002; Baker et al., 2007; Das et al., 2014), which has seen considerable recent work on English (Swayamdipta et al., 2017; Yang and Mitchell, 2017; Peng et al., 2018; Jiang and Riloff, 2021), there has not been any published work on Italian since the EVALITA-2011 shared task (Basili et al., 2013). Second, a clear perspective on how computational frame semantics can be useful in real-life applications is still missing.

We aim to advance the practical usability of frame semantics in Italian NLP in two ways. First, we test how well a recently developed multilingual model (LOME, Xia et al. (2021)) for FSP performs on Italian. For this purpose we use existing data from the EVALITA 2011 campaign, which is the only reference for Italian on FSP, as well as new "real world" data collected in the context of the socially relevant domain of femicides. Second, we show how frame semantics can be used in practice to run analysis on real world data. From both efforts, we draw some recommendations for practical developments in Italian FSP.

2 Semantic Frames for Events in Society

Frame semantics assumes that lexical units are points of access to complex conceptual structures: understanding the meaning of a word means to understand all of the knowledge that is associated with it. Every semantically loaded lexical item evokes a *frame*, a scenario-like unit of encyclopedic knowledge describing the concept associated to it. Frame semantics also describes the perspective in which the frame is seen. A classical example is that of a commercial transaction (Fillmore, 1971), where the same event can be presented either by foregrounding the buyer (e.g., *"Mary bought a book (from John)"*) or the seller (e.g., *"John sold a book (to Mary)"*). Perspectivization can be also related to syntactic constructions: an active sentence (*"Mary bought a book"*) and a passive one (*"The book has been bought"*) denote the same event, but make us access it via two different participants (Meluzzi et al., 2021).

It has been shown that the variability of linguistic expressions used to describe an event impacts the reader's perception of the event and its social significance. Previous work in psycholinguistics shows that in events involving violence (at any level), the linguistic backgrounding of agents hinders their responsibility and promote victim blaming (Huttenlocher et al., 1968; Bohner, 2001; Gray and Wegner, 2009; Zhou et al., 2021; Meluzzi et al., 2021). For instance, Te Brömmelstroet (2020) shows that media in the Netherlands frequently report on traffic crashes by foregrounding the more vulnerable participants (e.g., pedestrians or cyclists), while backgrounding car drivers. A similar pattern has been observed for news reports of femicides in Italy, where the victim tends to be foregrounded and the perpetrator backgrounded (Pinelli and Zanchi, 2021; Meluzzi et al., 2021).

While there have been some proposals to use frame semantics for analyzing media framing or applying it to social media texts (Ziem et al., 2018; Brambilla et al., 2019), we are not aware of previous work that applies frame semantics to the study of linguistic perspectivization of societal issues. We test this idea and present a preliminary analysis of how frames and syntactic constructions are used to perspectivize violence in a large corpus of femicide reports in the Italian press. We adopt the *data-to-text* approach to FrameNet analysis (Vossen et al., 2020; Remijnse and Minnema, 2020; Remijnse et al., 2021), where structured event metadata is linked to texts referencing real-world events. A crucial part of this method is defining *typical frames*, i.e., frames that are hypothesized to conceptualize important aspects of the targeted event type. For the femicide domain, we selected 15 typical frames;[1] some examples are in Table 1.

3 Frame Semantic Parsing for Italian

The shared task on Frame Labeling over Italian Texts (FLAIT) at EVALITA 2011 (Basili et al., 2013) introduce the only existing published Italian FSP models, as well as the only publicly available corpus for the task on generic texts. As shown in Table 2, the FLAIT corpus contains 1,569 annotated sentences, all of which are so-called *exemplars* containing a single annotated predicate and frame structure. Compared to the English Berkeley FrameNet (BFN), which contains also fully annotated documents, the models presented at FLAIT are impressive (scores up to 80%).

3.1 LOME experiments

LOME (Xia et al., 2021) is a recent end-to-end FSP model that reports excellent frame detection scores on English, and, thanks to its XLM-R encoder (Conneau et al., 2020), is the first cross-lingual FSP model, even though it was trained on English data only. Here, we propose several strategies for adapting LOME to Italian and making maximum use of the available data.

Strategies The simplest strategy, ***LOME-EN***, is to use the English-trained model in a zero-shot setup to make predictions for Italian texts. A downside of this approach is that the model is not able to tag the Italian-specific frames that have been created in the IFrameNet project (Basili et al., 2017), which also makes the evaluation on FLAIT data more challenging. FLAIT contains 10 frames that do not currently exist in BFN (7.4% of training instances and 6.0% of test instances). It therefore makes sense to also train LOME on FLAIT directly. In ***IT-Simple***, we only train on FLAIT data; in ***IT-Concat***, we train on the concatenation of FLAIT and the fully annotated documents from BFN; and in ***IT-Berkeley***, we train only on FLAIT but initialize the encoder with the parameters of LOME-EN.

Evaluation For use in real-life applications, what truly matters is *end-to-end* performance, i.e. from raw texts to the predictions of all predicate frames and associated roles. Full end-to-end evaluation is impossible in FLAIT since only one predicate per sentence is annotated. However, we can approximate it by obtaining the full predictions from the models and then evaluate only on FLAIT gold predicates. In this way, models are penalized for missing predicates that should have been annotated (but not for overgeneration). We use the *SeqLabel* metric (Minnema and Nissim, 2021) for scoring frame and role label predictions on a token-by-token basis.

Additionally, to test LOME against the 2011 models, we reimplement the FLAIT evaluation metrics, in which models are asked to predict (i) frames given a predicate (*Frame Detection* [FD]), (ii) semantic role spans given a frame (*Boundary*

[1] ABUSING, ATTACK, CAUSATION, CAUSE_HARM, CAUSE_MOTION, DEAD_OR_ALIVE, DEATH, EMOTION_DIRECTED, EVENT, EXPERIENCE_BODILY_HARM, HIT_TARGET, KILLING, QUARRELING, RAPE, USE_FIREARM.

Frame	Description	Example
KILLING	an agent *(Killer)* actively causes the death of a patient *(Victim)*	[The man] **killed** [his wife]
DEATH	someone *(Protagonist)* dies	[The woman] **died**
DEAD_OR_ALIVE	state of someone *(Protagonist)* being dead or alive	[She] was found **dead**
CAUSE_HARM	an agent *(Agent)* actively causes a patient *(Victim)* to be hurt	[He] **stabbed** [his girlfriend]
EVENT	an unspecified event *(Event)* happens	[The dramatic events] **happened** last week

Table 1: Examples of FrameNet frames relevant for describing femicides. Semantic role names indicated in *italics*, lexical units indicated in **bold**.

		English	Italian
fulltext	sentences	5,093	0
	frame instances	29,359	0
exemplar	sentences	163,801	1,569
	frame instances	169,473	1,569
total	sentences	168,894	1,569
	frame instances	198,832	1,569

Table 2: Sentences and annotations in the English and Italian datasets.

		frames			roles		
		P	**R**	**F**	**P**	**R**	**F**
EN	**LOME-EN**	0.89	0.70	0.78	0.69	0.59	0.64
IT	**LOME-EN**	0.63	0.52	0.57	0.63	0.50	0.56
	IT-Simple	-0.14	0.14	-0.01	-0.14	0.16	0.00
	IT-Concat	0.21	0.14	0.17	0.10	0.08	0.09
	IT-Berkeley	-0.07	0.17	0.05	0.04	0.12	0.09

Table 3: SeqLabel scores for gold predicates. Blue: baseline, green/red: performance deltas

Detection [BD]), or (iii) semantic role labels given a frame and the role spans (*Argument Classification* [AC]).[2]

Implementation We kept LOME model and training settings the same as described by Xia et al. (2021). During testing, we noticed that 56 instances in the FLAIT test set had misspelled frame labels,[3] causing a large drop in scores. We fixed these labels, but since we do not know if the original evaluation script also did this, we report the uncorrected scores in our GitHub repository.

Results Sequence labeling performance is reported in Table 3. The zero-shot LOME-EN model achieves an F1 score of 0.57 for frames and 0.56 for roles, substantially less than IT-Concat, which gets close to scores on English (0.74 F1 on frames, 0.63 on roles). The other two Italian models have mixed results, with improvements on recall but not on precision. However, IT-Berkeley outperforms both LOME-EN and IT-Simple, showing that reusing encoder weights helps boost performance.

Turning to EVALITA-style evaluation, in Table 4[4] we compare LOME against the best system from 2011, which is based on a SVM with a tree kernel (Croce et al., 2013). The most striking result is that, on frame prediction, the 2011 winner is still king, with the LOME-EN and IT-Concat models falling short by 0.24 and 0.04 points, respectively. For semantic role prediction, results are mixed: LOME-EN has a modest but consistent improvement on both span (BD) and label (AC) prediction, while IT-Concat improves on some setups but not on others.

3.2 Evaluating Real-World Performance

We explore how robust are our models when deployed on other data. We focus on frame prediction only, a task know to be harder to adapt across domains (Hartmann et al., 2017)

Femicide annotation We deployed the LOME-EN and IT-Concat on a set of femicide news reports (see §4) with typical frames (see §2) in an end-to-end setup (i.e., without predicates as input). Out of 4,444 frame predictions, the two models disagreed in 58% of cases. Next, for a subset of 150 conflicts, we manually annotated[5] which of the two predictions is better. Table 6 shows that LOME-EN performs much better than IT-Concat, especially on two of the most frequent typical frames (KILLING and EMOTION_DIRECTED). This is largely due to predicate detection: 47% of cases where LOME-EN is better than IT-Concat are due to IT-Concat not detecting the predicate; in conflicts for predicates that both models detected, IT-Concat slightly out-

[2] As we were unable to access the original evaluation script, we have attempted to reproduce it as faithfully as possible from the description in Basili et al. (2013).

[3] In these frame names, dashes were used in place of underscores, e.g. CAUSE-HARM instead of CAUSE_HARM.

[4] We only report strict scores for BD and AC. Full tables with token-based scores are in our GitHub repository.

[5] Annotation was done by a single annotator, who is also one of the co-authors of this paper. Annotation was blind and randomized, i.e., the annotator had no way to guess which prediction came from which model.

	run 1			run 2			run 3		
	P	R	F	P	R	F	P	R	F
FD									
2011-best	0.81	0.81	0.81	-	-	-	-	-	-
LOME-EN	-0.24	-0.24	-0.24	-	-	-	-	-	-
IT-Concat	-0.04	-0.04	-0.04	-	-	-	-	-	-
BD (strict)									
2011-best	0.67	0.73	0.69	0.67	0.73	0.69	-	-	-
LOME-EN	0.10	0.05	0.08	0.02	0.07	0.05	-	-	-
IT-Concat	-0.09	-0.06	-0.08	-0.10	-0.06	-0.08	-	-	-
AC (strict)									
2011-best	0.48	0.53	0.50	0.51	0.56	0.53	0.70	0.70	0.70
LOME-EN	-0.01	0.02	0.01	0.09	0.13	0.11	0.16	0.16	0.16
IT-Concat	-0.02	0.00	-0.01	-0.03	0.01	-0.01	0.14	0.14	0.14

Table 4: EVALITA-2011-style evaluation. As in the original task, run 1, 2, and 3 refer to predictions with, resp., no gold inputs, gold frame inputs, and gold frame and role span inputs.

	frames			
	all	IFN	BFN	fcd
FLAIT/dev				
num_examples	*123*	*123*	*113*	*14*
Simple SVM	0.59	0.59	0.60	0.71
LOME-EN	0.59	0.59	0.65	0.71
IT-Concat	0.85	0.85	0.87	0.93
FLAIT/test				
num_examples	*318*	*318*	*299*	*43*
Simple SVM	0.29	0.29	0.30	0.40
LOME-EN	0.57	0.57	0.60	0.60
IT-Concat	0.77	0.77	0.76	0.81
femicides				
num_examples	*43*	*43*	*43*	*43*
Simple SVM	0.14	0.14	0.14	0.14
LOME-EN	0.63	0.63	0.63	0.63
IT-Concat	0.72	0.72	0.72	0.72

Table 5: Generalizability scores

	best prediction			
	EN	IT	both	none
overall	0.51	0.12	0.12	0.25
non-null	0.17	0.22	0.44	0.17
by frame				
KILLING	0.70	0.19	0.11	0.00
EMOTION_D.	0.77	0.05	0.05	0.14
DEATH	0.33	0.05	0.19	0.42

Table 6: Conflict analysis on the femicides dataset. 'EN': LOME-EN; 'IT': IT-Concat; 'both'/'none': both models are equally correct/wrong.

performs LOME-EN. We speculate that this might be explained by the exemplar-style structure of the FLAIT corpus.

Generalization Table 5 shows frame detection scores on three evaluation sets: the FLAIT development set (10% held-out from the training set), the FLAIT test set, and the set of cases from our femicide annotation experiment in which at least one of the two models' predictions was marked as correct.[6] Since we do not have access to the original FLAIT models, we use a simple linear SVM,[7] trained on FLAIT, as an alternative baseline. The task is the same as the FLAIT FD task: the models are given the gold predicate and asked to predict the frame. Results are split by frame category: IFrameNet frames that FLAIT-trained models can be expected to know ('IFN'), BFN frames that LOME-EN can be expected to know ('BFN'), and typical frames for femicides ('fcd').

The results show several patterns that are relevant for real-world usability. First, both LOME models perform as good or better on typical femicide frames compared to other frames, which is a positive sign for the feasibility of our project. Furthermore, IT-Concat is clearly the overall best frame detection model, but only when it already knows which predicates to annotate (see above). However, it is also quite biased towards the FLAIT dataset, scoring substantially worse on the test and femicide datasets compared to the development set. By contrast, LOME-EN is very stable across datasets. The SVM baseline performs surprisingly well on the development set, but much worse on the test set and extremely poorly on the femicides dataset. We interpret this as a sign of the limited coverage of the FLAIT dataset, showing that good performance on the shared task is not necessarily indicative of real-world performance.

4 Frame-Based Analysis of Femicide News

In this section, we provide a concise overview of our initial work on applying frame semantic parsing to investigate news coverage of femicides.

Dataset We perform our analysis on a private dataset collected by the CRITS research team at RAI (Radiotelevisione Italiana) and made available for use in our project. The dataset contains 2,734 news articles from 31 different Italian news sources, reporting on 937 femicides perpetrated between 2015 and 2017, along with structured in-

[6] If the annotator indicated that both predictions for a particular predicate were equally good, we randomly selected one of the predictions as the 'gold' label.

[7] The SVM takes as input a bag-of-bigrams extracted from a context window of 5 tokens before and after the predicate.

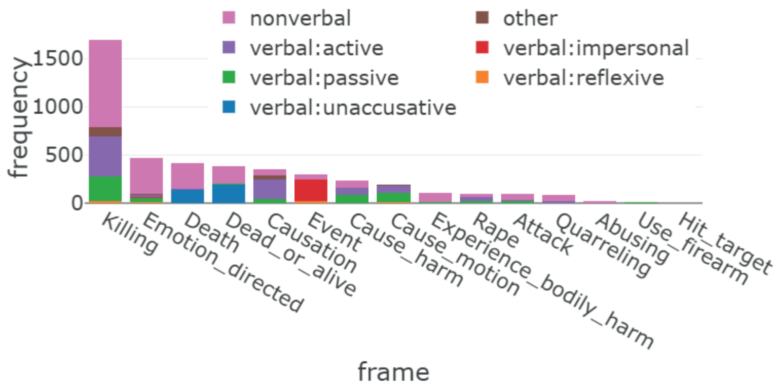


Figure 1: Typical frame frequencies, split by syntactic construction

formation about these femicides (Belluati, 2021)[8]. The dataset is unique because it includes rich event metadata, and contains various news article per femicide, allowing for investigating variation in framing of the same event along different dimensions, e.g., over time or by news source.

Analysis Based on our findings in §3, especially from the human evaluation experiment, we deploy the LOME-EN model to automatically annotate a randomly chosen 200K word subcorpus covering 10% of all events.The frame semantic annotations are enriched with dependency parses produced by `spaCy` (Honnibal et al., 2020), which are converted into syntactic construction annotations using a set of heuristics.

Figure 1 shows our main results. KILLING is by far the most frequent typical frame, followed by EMOTION_DIRECTED and DEATH. Looking at syntax, we find that *nonverbal* constructions, in which the predicate is expressed by a noun or adjective (e.g., *"l'omicidio"* "the murder") are dominant in many frames. Instead, *verbal:active* constructions (e.g., *"X uccide Y"* "X kills Y") are much rarer, as are *verbal:passive* (e.g., *"X è uccisa"* "X is killed") and *verbal:unaccusative* (e.g., *"X è deceduta"* "X has died").

Looking at semantic roles, patterns that vary greatly depending on frames and constructions. In general, semantic roles that are likely to refer to the perpetrator appear to be expressed much less frequently than those referring to the victim. For KILLING, 60% of all instances express a Victim role, while only 33% express a Killer role. However, instances with a nonverbal construction only express these roles in 40% and 20% of cases, respectively, against 71% and 87% in active constructions. On the other hand, DEATH expresses a victim-like role (Protagonist) in 79% of cases, whereas its only role that can encode a perpetrator (Explanation) occurs in 14% of cases.

While our analysis is too preliminary to draw strong conclusions, our findings are consistent with previous work: agentivity-backgrounding constructions (especially nonverbal) are very common, and semantic roles encoding the victim are more frequent than those encoding the perpetrator. What our frame analysis adds to previous work is information about the semantics of the analyzed constructions. For example, the dominance of KILLING suggests that femicides tend to be framed as agentive at least on a lexical level, even if the perpetrator is often backgrounded syntactically. On the other hand, non-agentive ways of framing the event (DEATH, DEAD_OR_ALIVE, EVENT) are also relatively common, accounting for 24% of frame instances.

5 Conclusions

We took initial steps towards addressing (i) the lack of recent frame semantic parsing models, and (ii) a missing perspective on how frame semantic analysis can be applied in practice. We adapted the multilingual LOME parser (Xia et al., 2021) to Italian, tested it against the EVALITA-2011 benchmark, and performed experiments to evaluate its real-world performance. Furthermore, we hypothesize that frame semantics can be a valu-

[8]The dataset has been collected as an outcome of the PRIN 2015 research project *Rappresentazioni sociali della violenza sulle donne: il caso del femminicidio in Italia*.

able analysis tool for analyzing backgrounding (and indirectly, blame attribution) of event participants, and propose news reports about femicides as an example of a domain where this type of analysis is very socially relevant.

Our results indicate that LOME-based models can achieve acceptable performance, both on the EVALITA benchmark and out-of-domain on femicide reports, even without a large quantity of training data. We also found that a cross-lingual approach is useful: training on the concatenation of English and Italian data yields substantial improvements over using only Italian data, and even a zero-shot approach with only English data works quite well. However, our real-world performance analysis highlights key limitations of the Italian data: while models trained on EVALITA can achieve good frame detection performance, they fail when used 'end-to-end', with predicate identification seemingly the main bottleneck.

Finally, we performed a preliminary framing analysis of a large dataset covering femicides in Italy. While our analysis method is still in very early stages, we believe that our initial results demonstrate that frame semantics is meaningful for analyzing femicides and other social issues, and that it complements earlier construction-based approaches. In the future, we aim to expand our analysis system to make it usable for different social applications: for example, one could envision systems that can help social scientists test specific hypotheses about media reporting, help activists identify and highlight biased forms of reporting, or help make journalists more aware of their writing and its possible social-cognitive effects.

Acknowledgements

We would like to thank the CRITS department at RAI for giving us access to the femicides dataset. We would also like to thank our collaborators in the broader responsibility framing research effort that this work is part of: Marion Bartl, Gaetana Ruggiero, Marco te Brömmelstroet, and Eva Kwakman. Authors G.M., T.C., and M.N. worked on this paper as part of the project *Framing situations in the Dutch language* (code: VC.GW17.083/6215), funded by the Dutch National Science Organization (NWO).

References

Collin F. Baker, Charles J. Fillmore, and Beau Cronin. 2003. The structure of the FrameNet database. *International Journal of Lexicography*, 16(3):281–296.

Collin Baker, Michael Ellsworth, and Katrin Erk. 2007. SemEval-2007 task 19: Frame semantic structure extraction. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 99–104, Prague, Czech Republic, June. Association for Computational Linguistics.

Roberto Basili, Diego De Cao, Alessandro Lenci, Alessandro Moschitti, and Giulia Venturi. 2013. Evalita 2011: The frame labeling over Italian texts task. In Bernardo Magnini, Francesco Cutugno, Mauro Falcone, and Emanuele Pianta, editors, *Evaluation of Natural Language and Speech Tools for Italian*, pages 195–204, Berlin, Heidelberg. Springer.

Roberto Basili, Silvia Brambilla, Danilo Croce, and Fabio Tamburini. 2017. Developing a large scale FrameNet for Italian: the IFrameNet experience. In Roberto Basili, Malvina Nissim, and Giorgio Satta, editors, *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017), Rome, Italy, December 11-13, 2017*, volume 2006 of *CEUR Workshop Proceedings*. CEUR-WS.org.

M. Belluati. 2021. *Femminicidio. Una lettura tra realtà e interpretazione*. Biblioteca di testi e studi. Carocci.

Gerd Bohner. 2001. Writing about rape: Use of the passive voice and other distancing text features as an expression of perceived responsibility of the victim. *British Journal of Social Psychology*, 40(4):515–529.

Silvia Brambilla, Alessio Palmero Aprosio, and Stefano Menini. 2019. BullyFrame: Cyberbullying meets FrameNet. In Raffaella Bernardi, Roberto Navigli, and Giovanni Semeraro, editors, *Proceedings of the Sixth Italian Conference on Computational Linguistics, Bari, Italy, November 13-15, 2019*, volume 2481 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Silvia Brambilla, Danilo Croce, Fabio Tamburini, and Roberto Basili. 2020. Automatic induction of FrameNet lexical units in Italian. In Johanna Monti, Felice Dell'Orletta, and Fabio Tamburini, editors, *Proceedings of the Seventh Italian Conference on Computational Linguistics, CLiC-it 2020, Bologna, Italy, March 1-3, 2021*, volume 2769 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8440–8451, Online, July. Association for Computational Linguistics.

Danilo Croce, Emanuele Bastianelli, and Giuseppe Castellucci. 2013. Structured kernel-based learning for the frame labeling over Italian texts. In *Evaluation of Natural Language and Speech Tools for Italian*, pages 195–204, Berlin, Heidelberg. Springer Berlin Heidelberg.

Dipanjan Das, Desai Chen, André F. T. Martins, Nathan Schneider, and Noah A. Smith. 2014. Frame-semantic parsing. *Computational Linguistics*, 40(1):9–56, March.

Charles J. Fillmore. 1971. Subjects, speakers, and roles. *Synthese*, 21(3-4).

Charles J. Fillmore. 1985. Frames and the semantics of understanding. *Quaderni di semantica*, 6(2):222–254.

Charles J. Fillmore. 2006. Frame semantics. In D. Geeraerts, editor, *Cognitive Linguistics: Basic Readings*, pages 373–400. De Gruyter Mouton, Berlin, Boston. Originally published in 1982.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

Kurt Gray and Daniel M. Wegner. 2009. Moral typecasting: divergent perceptions of moral agents and moral patients. *Journal of Personality and Social Psychology*, 96:505–520.

Silvana Hartmann, Ilia Kuznetsov, Teresa Martin, and Iryna Gurevych. 2017. Out-of-domain FrameNet semantic role labeling. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 471–482, Valencia, Spain, April. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Janellen Huttenlocher, Karen Eisenberg, and Susan Strauss. 1968. Comprehension: Relation between perceived actor and logical subject. *Journal of Verbal Learning and Verbal Behavior*, 7:527–530.

Tianyu Jiang and Ellen Riloff. 2021. Exploiting definitions for frame identification. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2429–2434, Online, April. Association for Computational Linguistics.

Alessandro Lenci, Martina Johnson, and Gabriella Lapesa. 2010. Building an Italian FrameNet through semi-automatic corpus analysis. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Chiara Meluzzi, Erica Pinelli, Elena Valvason, and Chiara Zanchi. 2021. Responsibility attribution in gender-based domestic violence: A study bridging corpus-assisted discourse analysis and readers' perception. *Journal of Pragmatics*, 185:73–92.

Gosse Minnema and Malvina Nissim. 2021. Breeding Fillmore's chickens and hatching the eggs: Recombining frames and roles in frame-semantic parsing. In *Proceedings of the 14th International Conference on Computational Semantics*. `https://iwcs2021.github.io/proceedings/iwcs/pdf/2021.iwcs-1.15.pdf`.

Hao Peng, Sam Thomson, Swabha Swayamdipta, and Noah A. Smith. 2018. Learning joint semantic parsers from disjoint data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1492–1502, New Orleans, Louisiana, June. Association for Computational Linguistics.

Erica Pinelli and Chiara Zanchi. 2021. Gender-based violence in italian local newspapers: How argument structure constructions can diminish a perpetrator's responsibility. *Discourse Processes between Reason and Emotion: A Post-disciplinary Perspective*, page 117.

Levi Remijnse and Gosse Minnema. 2020. Towards reference-aware FrameNet annotation. In *Proceedings of the International FrameNet Workshop 2020: Towards a Global, Multilingual FrameNet*, pages 13–22, Marseille, France, May. European Language Resources Association.

Levi Remijnse, Marten Postma, and Piek Vossen. 2021. Variation in framing as a function of temporal reporting distance. In *Proceedings of the 14th International Conference on Computational Semantics*. `https://iwcs2021.github.io/proceedings/iwcs/pdf/2021.iwcs-1.22.pdf`.

Swabha Swayamdipta, Sam Thomson, Chris Dyer, and Noah A. Smith. 2017. Frame-semantic parsing with softmax-margin segmental rnns and a syntactic scaffold. *CoRR*, abs/1706.09528.

Marco Te Brömmelstroet. 2020. Framing systemic traffic violence: Media coverage of dutch traffic crashes. *Transportation research interdisciplinary perspectives*, 5.

Sara Tonelli and Emanuele Pianta. 2008. Frame information transfer from English to Italian. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May. European Language Resources Association (ELRA).

Sara Tonelli, Daniele Pighin, Claudio Giuliano, and Emanuele Pianta. 2009. Semiautomatic Development of FrameNet for Italian. In *Proceedings of the FrameNet Workshop and Masterclass, Co-located with the Seventh International Workshop on Treebanks and Linguistic Theories*.

Piek Vossen, Filip Ilievski, Marten Postma, Antske Fokkens, Gosse Minnema, and Levi Remijnse. 2020. Large-scale cross-lingual language resources for referencing and framing. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3162–3171, Marseille, France, May. European Language Resources Association.

Patrick Xia, Guanghui Qin, Siddharth Vashishtha, Yunmo Chen, Tongfei Chen, Chandler May, Craig Harman, Kyle Rawlins, Aaron Steven White, and Benjamin Van Durme. 2021. LOME: Large ontology multilingual extraction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 149–159, Online, April. Association for Computational Linguistics.

Bishan Yang and Tom Mitchell. 2017. A joint sequential and relational model for frame-semantic parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256, Copenhagen, Denmark, September. Association for Computational Linguistics.

Karen Zhou, Ana Smith, and Lillian Lee. 2021. Assessing cognitive linguistic influences in the assignment of blame. In *Proceedings of the Ninth International Workshop on Natural Language Processing for Social Media*, pages 61–69, Online, June. Association for Computational Linguistics.

Alexander Ziem, Christian Pentzold, and Claudia Fraas. 2018. Medien-Frames als semantische Frames: Aspekte ihrer methodischen und analytischen Verschränkung am Beispiel der 'Snowdon-Affäre' [Media frames as semantic frames: Aspects of their methodological and analytical entanglement in the example of the Snowdon affair]. In Detmer Wulf Alexander Ziem, Lars Inderelst, editor, *Frames interdisziplinär: Modelle, Anwendungsfelder, Methoden*, pages 155–184, Düsseldorf. DUP.

Leveraging Bias in Pre-Trained Word Embeddings for Unsupervised Microaggression Detection

Tolúlọpẹ́ Ògúnrẹ̀mí[1], **Nazanin Sabri**[2], **Valerio Basile**[3], **Tommaso Caselli**[4]
1. Stanford University, United States, tolulope@stanford.edu
2. Independent Researcher nazanin.sabrii@gmail.com
3. University of Turin, Italy, valerio.basile@unito.it
4. University of Groningen, Netherlands, t.caselli@rug.nl

Abstract

Microaggressions are subtle manifestations of bias (Breitfeller et al., 2019). These demonstrations of bias can often be classified as a subset of abusive language. However, not as much focus has been placed on the recognition of these instances. As a result, limited data is available on the topic, and only in English. Being able to detect microaggressions without the need for labeled data would be advantageous since it would allow content moderation also for languages lacking annotated data. In this study, we introduce an unsupervised method to detect microaggressions in natural language expressions. The algorithm relies on pre-trained word-embeddings, leveraging the bias encoded in the model in order to detect microaggressions in unseen textual instances. We test the method on a dataset of racial and gender-based microaggressions, reporting promising results. We further run the algorithm on out-of-domain unseen data with the purpose of bootstrapping corpora of microaggressions "in the wild", and discuss the benefits and drawbacks of our proposed method.

1 Introduction

The growth of Social Media platforms has been accompanied by an increased visibility of expressions of socially unacceptable language online. In a 2016 Eurobarometer survey, 75% of people who follow or participate in online discussions have witnessed or experienced abuse or hate speech. With this umbrella term, different phenomena can be identified ranging from offensive language to more complex and dangerous ones, such as hate speech or doxing. Recently, there has been a growing interest by the Natural Language Processing community in the development of language resources and systems to counteract socially unacceptable language online. Most previous work has focused on few, easy to model phenomena, ignoring more subtle and complex ones, such as microaggressions (Jurgens et al., 2019).

Microaggressions are brief, everyday exchanges that denigrate stigmatised and culturally marginalised groups (Merriam-Webster, 2021). They are not always perceived as hurtful by either party, and they can often be detected as positive statements by current hate-speech detection systems (Breitfeller et al., 2019). The occasionally unintentional hurt caused by such comments is a reflection of how certain stereotypes of others are baked into society. Sue et al. (2007) define microaggressions in the racial context, particularly when directed toward people of color, as "brief and commonplace daily verbal, behavioral, or environmental indignities", such as: "you are a credit to your race." (intended message: it is unusual for someone of your race to be intelligent) or "do you think you're ready for college?" (indented message: it is unusual for people of color to succeed). The need for moderation of hateful content has previously been explored. For instance, Mathew et al. (2019b) analyses the temporal effects of allowing hate speech on Gab, and finds that the language of users tends to become more and more similar to that of hateful users over time. Mathew et al. (2019a) further highlights that the spreading speed and reach of hateful content is much higher than with the non-hateful content. As a result, being able to remove instances of hateful language, such as microaggressions, is of great importance.

Previous work on microaggressions with computational methods is quite recent. Breitfeller et al. (2019) is one of the first work to address microaggressions in a systematic way, also introducing a first dataset, SelfMA. A further contribution specifically focused on racial microaggression is Ali et al. (2020), where the authors focus on the development of machine learning systems.

In this study we introduce an unsupervised method for microaggression detection. Our method utilizes the existing bias in word-embeddings to detect words with biased connotations in the message. Although unsupervised approaches tend to be less competitive than their supervised counterparts, our method is language-independent and thus it can be applied to any language for which embedding representations exist. Furthermore, the reliance of our methods on specific lexical items and their context of occurrence makes transparent the flagging of a message as an instance of a microaggression. In addition to the usefulness of our method in languages with no labeled data, the reliance of our model on words in the sentences would make it interpretable as it allow human moderators to understand what the system has based its decision on.

Our contributions can be summarised as follows:

- we introduce a **new unsupervised method** for the detection of microaggressions which builds on top of pre-trained word embeddings;
- we **compare the performance** of our model using different pre-trained word embeddings (Glove, FastText, and Word2Vec) and discuss the potential reasons behind the differences;
- we **test** the proposed algorithm **on unseen data from a different domain** (i.e., Twitter), in order to qualitatively evaluate its efficacy in discovering new instances of microaggression.

The rest of this paper is structured as follows: we introduce our method in Section 2. The data and our results are reported in Section 3. We deploy our model and discuss its limitations in Section 4. Finally, we present the conclusion and future work in Section 5.

2 Use the Bias Against the Bias

Embedded representations, either from pre-trained word embeddings or pre-trained language models, have been shown to contain and amplify the biases present in the data used to generate them (Bolukbasi et al., 2016; Lauscher and Glavaš, 2019; Bhardwaj et al., 2020). As such, they often exhibit gender and racial bias (Swinger et al., 2019). Many studies have attempted to reduce this bias (Yang and Feng, 2020; Zhao et al., 2018; Manzini et al., 2019). In this work, we take a different turn by using this bias to our advantage: rather than taming the hurtfulness of the representations (Schick et al., 2021), we actively use it to promote social good. In this first study, we employ word representations derived from generic textual corpora of English, in order to capture the background knowledge needed to disambiguate instances of microaggressions in the text. Recently, however, there have been studies involving word representations created from tailored collections of social media content aimed at capturing abusive phenomena like verbal aggression (Dynel, 2021) and hate speech (Caselli et al., 2020).

We devise a simple and effective method that exploits existing bias in word embeddings and identify words in a message that are related to particular and distant semantic areas in the embedding space. Messages are analysed in three steps: first, for each token t^i we compute its relatedness to a list of manually curated seed words $s = s_1, ..., s_n$ denoting potential targets of microaggressions; second, we consider only the similarities of the pairs (t_i, s_j) above an empirical *similarity threshold* ST and compute their variance v_i; finally, we classify the token t_i as a micro aggression trigger, and consequently the message as a micro aggression, if the v_i is above an empirically determined *variance threshold* VT.

The intuitive idea behind this algorithm is that some lexical elements in a verbal microaggression are often (yet sometimes subtly) hinting at specific features of the recipient of the message, in an otherwise neutral lexical context.

In this work, we choose to focus on microaggressions related to race and gender, therefore the seed words have to be chosen accordingly. The seed word lists for race and gender are, respectively, *[white, black, asian, latino, hispanic, arab, african, caucasian]* and *[girl, boy, man, woman, male, female]* for gender. There is also a practical reasons to focus on gender and race, namely the scarcity of data available for other categories of microaggression and other idiosincrasies of the

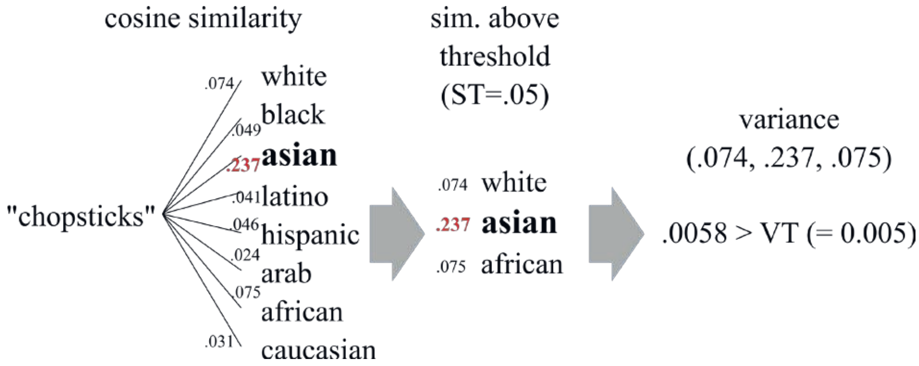


Figure 1: Worked example of unsupervised method for word ”chopsticks” in the message *”Ford: Built With Tools, Not With Chopsticks”*

available datasets — the religion class was specific to different religions, therefore hard to generalise, sexuality and gender presented a large overlap, and so on.

An example of how the proposed method works is illustrated in Figure 1. In the example, consider the word ”chopsticks” in the message *”Ford: Built With Tools, Not With Chopsticks”* (from the SelfMA dataset, described in Section 3). The target word exhibits a much higher relatedness to the word *asian* (0.237) than any other seed words. Even just considering the seed words with a similarity above a fixed threshold (*white*, *asian* and, *african*), the variance of their similarity score with respect to *chopsticks* is still higher than the variance threshold, and therefore this target word, in this context, triggers a microaggression according to the algorithm. This process is repeated for all the words in the message in order to detect microaggressions. Some categories of words are bound to exhibit a high relatedness to all the seed words, e.g., “people” or “human”. This is the reason to introduce the variance threshold in the final step of our algorithm, to filter out these cases when classifying a given message, and instead focus on words that are related to different races (or genders) unevenly, with a skewed distribution of similarity scores.

An important by-product of this algorithm is that the output is one or more trigger words, in addition to the microaggression label — in the example, the trigger word is indeed *chopsticks* — therefore enabling a more informative and interpretable decision process.

Source	Number of posts
SelfMA Gender	1,314
SelfMA Racial	1,278
Tumblr	2,021

Table 1: Statistics of the two subsets of the SelfMA dataset used in this paper, and the extra data downloaded to balance the dataset.

3 Experiments

To test our method, we use two subsets of the *SelfMA: microaggressions.com* dataset (Breitfeller et al., 2019), comprised of 1,314 and 1,278 Tumblr posts respectively[1]. The posts in SelfMA are all instances of microaggressions, manually tagged with one of four categories: race, gender, sexuality and religion. These posts can be tagged with more than one form of microaggressions, meaning certain instances can appear in both subsets of race and gender used for the purposes of this study. The dataset consists of first and second hand accounts of microaggressions, as well as direct quotes of phrases or sentences said to the person posting. In order to reduce linguistic perturbation introduced by accounts of a situation, we only take direct quotes found in the dataset as instances of microaggressions that we can detect with our unsupervised method. For training, we pull out direct quotes from the gender (561) and racial (519) dataset to test the algorithm. In order to balance the dataset, we scraped 2,021 random Tumblr posts, for a total of 4,612 instances. Table 1 summarises the composition of our dataset.

It is important to note that a microaggression can have multiple tags, so there is an overlap of

[1]Tumblr is a popular American microblogging platform `https://www.tumblr.com`

instances. However, the seed words used to detect microaggression types in the method are different for each target phenomenon (e.g., race, gender).

We ran the algorithm on the *SelfMA* dataset, empirically optimising the two thresholds on the training split, for each word embedding type and each microaggression category, filtering by the seed words listed in Section 2. We test the algorithm with three pre-trained word embedding models for English, namely `FastText` (Joulin et al., 2016) (trained on Wikipedia and Common Crawl), `word2vec` (Mikolov et al., 2013) (trained on Google News), and `GloVe` (Pennington et al., 2014) (trained on Wikipedia, GigaWord corpus, and Common Crawl). The optimization is performed by exhaustive grid search over the hyperparamter space.

The results, shown in Table 2, indicate that `FastText` has a better F1 score on Racial microaggressions while `word2vec` performs better on Gender microaggressions. The difference in performance between `FastText` and `word2vec` is not major, and we attribute this to the difference between the corpora on which the two models were trained (i.e., web crawl and Wikipedia for `FastText` vs. news data for `word2vec`). The `GloVe` pretrained model, trained on a combination of newswire texts, encyclopedic entries and texts from the Web, underperforms in both experiments. In general, the absolute figures are encouraging, especially considering the simplicity of this unsupervised approach.

4 Discovering Microaggressions

To better understand the performance of our unsupervised model, we performed an additional experiment. Our goal is to understand the false positive results and the potential harm the model could cause. To do so, we use our unsupervised model to label unseen instances from another domain (Twitter) than the SelfMA dataset (Tumblr) in order to see how the model would perform in detecting microaggressions.

We begin by performing keyword searches on Twitter (using Twitter's official API) and collect a new dataset of of 3M tweets with seven keywords potentially containing race and gender expressions.Next, we set the threshold values ST and VT in our model in order to obtain the highest Precision scores, rather than the highest F1 value. This step is performed exactly like the optimization described in Section 2 with the only difference of the target metric. The aim of this step is to only label tweets as microaggressions with the highest possible degree of confidence. We set $ST = 0.12$ and $VT = 0.014$ for racial microaggressions leading to Precision of .931 and $ST = 0.13$ and $VT = 0.019$ for gender-based microaggressions leading to a Precision of .912. Precision has been measured on the original SelfMA dataset used as a validation set.

We then run the unsupervised model on the new Twitter dataset by automatically labelling 256,843 tweets for gender and 373,631 tweets for race. After the data is labeled, we manually explore the positive instances in order to evaluate the performance of the model. The algorithm tuned for high precision found in this dataset 6,306 gender-related microaggression candidates, 13,004 race-related microaggression candidates.

We find that while the model does detect actual instances of microaggression, there is a noticeable amount of false positive instances. These tweets discuss race or gender in some manner. However, they do not necessarily contain microaggressions towards these groups. While the model does learn to detect discussions of these topics, it seems to sometimes confuse these discussions with microaggressions towards the aforementioned groups. Some examples follow, paraphrased to avoid tracking the original messages.

> *Saying "Arrested Development isn't funny" in an office full of women just to feel something*

> *"Men have moustaches, women have oversized bracelets"*

The humorous attempts in this tweets hinge on gender stereotypes, and therefore in some contexts it could be perceived as offensive by some recipients. The high relatedness in the word embedding space between some words (moustaches and bracelets) and gender-related seed words (men and women) triggers the detection algorithm.

The automatic detection of racial microaggressions "in the wild" is more challenging than gender-based ones, according to our manual exploration of this automatically labeled dataset. This may be due to the difficulty of crafting a list of seed words that is sufficiently race-related, but at the same time avoids generating too many false positives. We indeed found many of them,

Target	Model	Class	Precision	Recall	F1-Score
Gender	`FastText`	not-MA	.609	.746	.671
		MA	.714	.570	.634
		macro avg.			.680
	`GloVe`	not-MA	.692	.380	.491
		MA	.603	.848	.705
		macro avg.			.598
	`word2vec`	not-MA	.659	.789	.718
		MA	.769	.634	.694
		macro avg.			**.706**
Race	`FastText`	not-MA	.659	.875	.654
		MA	.814	.547	.752
		macro avg.			**.702**
	`GloVe`	not-MA	.765	.371	.500
		MA	.611	.896	.726
		macro avg.			.613
	`word2vec`	not-MA	.640	.814	.747
		MA	.776	.584	.667
		macro avg.			.692

Table 2: Results of the experiment on the Gender and Racial subset of SelfMA, in terms of Precition (P), Recall (R), and F1-score (F1) on the positive class (MA), on the negative class (not-MA), and their macro-average. Best scores per microagression category are in bold.

mainly due to named entities and multi-word expressions such as "White House", or simply because of the polysemy of color words, e.g. "black" and "white". We, however, still found instances of messages containing different extent of racial stereotyping.

> *"why are you being so dramatic? just say I'm not originally arab, you don't have to fight about it"*

> *"I will need to explain that to the chinese old lady who works at my school's administrative office"*

In summary, running the unsupervised microaggression detection algorithm on unseen data seems to represent a promising intermediate step towards the semi-automatic creation of language resources for this phenomenon. While the accuracy is not ideal, and lists of seed words have to be handcrafted carefully in order to avoid false positives, these drawbacks are balanced by the fairly cheap computational cost and the ease of application in a multilingual scenario.

5 Conclusion and Future Work

In this paper we introduce a novel algorithm that exploits the existing bias in pre-trained word embeddings to detect subtly abusive language phenomena such as microagressions. While supervised methods of detection in the field of natural language processing are plentiful, these methods are only viable for languages and topics with available labeled datasets. That is however not the case for many languages. As a result, the unsupervised method of detection introduced in this study could help address the need for the moderation of microaggressions in languages other than English. This is further helped by the availability of multilingual word-embeddings as they would allow the method to be used in any of the languages supported by the embedding.

The method is unsupervised and only needs a small list of seed words. Considering its simplicity, the results obtained from an experiment on a dataset of manually annotated microaggressions are very promising. Further, the method is transparent, explicitly identifying the words triggering a microaggression, and thus paving the way for explainable microaggression detection.

Although the preliminary results are promising, an experiment on unseen data from a different domain shows that there is leeway for improvement. Given that we are looking at the explicit words used in each message, our method is not sensitive to implicit expressions like "you people" or "your kind", often occurring in microaggressions. We would have to add further steps to our algorithm to catch expressions like these.

Polysemy is another known issue, e.g., in words like "black" and "white" whose relatedness to certain identified trigger words could not necessarily be due to race. While a careful composition of the seed word lists helps to minimize this issue, a systematic approach to polysemy would certainly be desirable. The seed word list may also be expanded, either manually or exploiting existing lexicons such as HurtLex (Bassignana et al., 2018) for offensive terms (including stereotypes for several categories of individuals) or specialized lists of identity-related terms[2].

In future work, we plan on improving our model to account for lexical ambiguity, and the complexity derived from the interference between pragmatic phenomena and aggression, e.g., in humorous and ironic messages, following the intuition in recent literature (Frenda, 2018) about the interconnection between irony or sarcasm and abusive language online. Our current plan is to apply the algorithm presented in this paper to bootstrap the creation of a multilingual resource of online verbal microaggressions and release it to the research community.

Acknowledgements

This work of Valerio Basile is partially funded by the project "Be Positive!" (under the 2019 "Google.org Impact Challenge on Safety" call).

References

Omar Ali, Nancy Scheidt, Alexander Gegov, Ella Haig, Mo Adda, and Benjamin Aziz. 2020. Automated detection of racial microaggressions using machine learning. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 2477–2484. IEEE.

Elisa Bassignana, Valerio Basile, and Viviana Patti. 2018. Hurtlex: A multilingual lexicon of words to hurt. In *5th Italian Conference on Computational Linguistics, CLiC-it 2018*, volume 2253, pages 1–6. CEUR-WS.

Rishabh Bhardwaj, Navonil Majumder, and Soujanya Poria. 2020. Investigating gender bias in bert. *arXiv preprint arXiv:2009.05021*.

Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *arXiv preprint arXiv:1607.06520*.

Luke Breitfeller, Emily Ahn, David Jurgens, and Yulia Tsvetkov. 2019. Finding microaggressions in the wild: A case for locating elusive phenomena in social media posts. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1664–1674.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. 2020. HateBERT: Retraining BERT for Abusive Language Detection in English. *arXiv preprint arXiv:2010.12472*.

Marta Dynel. 2021. Humour and (mock) aggression: Distinguishing cyberbullying from roasting. *Language & Communication*, 81:17–36.

Simona Frenda. 2018. The role of sarcasm in hate speech. a multilingual perspective. In *e Doctoral Symposium of the XXXIVInternational Conference of the Spanish Society for Natural Language Processing (SEPLN 2018)*, pages 13–17. Lloret, E.; Saquete, E.; Martínez-Barco, P.; Moreno, I.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

David Jurgens, Libby Hemphill, and Eshwar Chandrasekharan. 2019. A just and comprehensive strategy for using NLP to address online abuse. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3658–3666, Florence, Italy, July. Association for Computational Linguistics.

Anne Lauscher and Goran Glavaš. 2019. Are we consistently biased? multidimensional analysis of biases in distributional word vectors. *arXiv preprint arXiv:1904.11783*.

Thomas Manzini, Yao Chong Lim, Yulia Tsvetkov, and Alan W Black. 2019. Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. *arXiv preprint arXiv:1904.04047*.

Binny Mathew, Ritam Dutt, Pawan Goyal, and Animesh Mukherjee. 2019a. Spread of hate speech in online social media. In *Proceedings of the 10th ACM conference on web science*, pages 173–182.

Binny Mathew, Anurag Illendula, Punyajoy Saha, Soumya Sarkar, Pawan Goyal, and Animesh

[2]See for instance this compendium of LGBTQIA+ terminology: `https://www.umass.edu/stonewall/sites/default/files/documents/allyship_term_handout.pdf`

Mukherjee. 2019b. Temporal effects of unmoderated hate speech in gab. *arXiv preprint arXiv:1909.10966*.

Merriam-Webster. 2021. Merriam-webster's definition of microaggression. https://www.merriam-webster.com/dictionary/microaggression. Accessed: 2021-03-08.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Jeffrey Pennington, R. Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Timo Schick, Sahana Udupa, and Hinrich Schütze. 2021. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in nlp. *arXiv preprint arXiv:2103.00453*.

Derald Sue, Christina Capodilupo, Gina Torino, Jennifer Bucceri, be Aisha, Kevin Nadal, and Marta Esquilin. 2007. Racial microaggressions in everyday life: Implications for clinical practice. *The American psychologist*, 62:271–86, 05.

Nathaniel Swinger, Maria De-Arteaga, Neil Thomas Heffernan IV, Mark DM Leiserson, and Adam Tauman Kalai. 2019. What are the biases in my word embedding? In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 305–311.

Zekun Yang and Juan Feng. 2020. A causal inference method for reducing gender bias in word embedding relations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9434–9441.

Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018. Learning gender-neutral word embeddings. *arXiv preprint arXiv:1809.01496*.

REDIT: A Tool and Dataset for Extraction of Personal Data in Documents of the Public Administration Domain

Teresa Paccosi, Alessio Palmero Aprosio
Fondazione Bruno Kessler
Via Sommarive 18, Trento (Italy)
[tpaccosi|aprosio]@fbk.eu

Abstract

English. New regulations on transparency and the recent policy for privacy force the public administration (PA) to make their documents available, but also to limit the diffusion of personal data. The present work displays a first approach to the extraction of sensitive data from PA documents in terms of named entities and semantic relations among them, speeding up the process of extraction of these personal data in order to easily select those which need to be hidden. We also present the process of collection and annotation of the dataset.

Italiano. *Le nuove regolamentazioni sulla trasparenza e la recente legislazione sulla privacy hanno spinto la pubblica amministrazione a rendere i loro documenti pubblicamente consultabili limitando però la diffusione di dati personali. Presentiamo qui un primo approccio all'estrazione di questi dati da documenti amministrativi in termini di named entities e relazioni semantiche tra di esse, in modo da facilitare la selezione dei dati che devono rimanere privati. Presentiamo inoltre il processo di collezione e annotazione del dataset.*

1 Introduction

In recent years, public administrations (PA) in the Italian government have been forced to publish a huge amount of documents, to make them available to citizens, organisations, and authorities. This is the result of the recent legislation about the transparency. For instance, municipalities have to share their documents in a virtual place called *Albo Pretorio*. In most cases, the online publication of these acts is a necessary condition for their purposes to become effective.[1]

On the other side, the General Data Protection Regulation (GDPR), approved in 2016 by the European Union, enhances individuals' control and rights over their personal data, limiting its diffusion over any medium (especially including online platforms such as websites and social networks).

In this context, it is important for the public servants within the PA to amend some documents by hiding the data that cannot be publicly published. Nowadays, most of this work is done manually, hiding the sensitive information document by document. This procedure is clearly time-consuming, non-scalable, and error-prone.

Natural Language Processing (NLP) techniques can be seen as a watershed between a manual management of the PA documents and a new generation of instruments that will finally speed up the process, leaving manual effort as the sole final check just before the publication of the data.

This is not the first time this problem is tackled using NLP, but past works are mainly focused on English and limited to the entity extraction task (Guo et al., 2021).

Our approach to the extraction of personal data from documents focuses on a combination of three NLP instruments:

- **Named-entity Recognition (NER).** This task consists in seeking texts in natural language to locate and classify named entities (NE) mentioned in them. This search is usually limited to a few needed categories: the most common are persons, locations, and organisations. Several approaches have been

[1] In the Italian legislation, this is called "referto di pubblicazione". See also: `http://qualitapa.gov.it/`

used in literature, between completely rule-based (Appelt et al., 1993; Budi and Bressan, 2003) and machine learning-based (Chiu and Nichols, 2016; Strubell et al., 2017; Devlin et al., 2019), including some hybrid approaches, for example using gazettes of known entities belonging to a particular category (Finkel et al., 2005). In this paper, we use the last approach, mixing a Conditional Random Fields (CRF) algorithm (Lafferty et al., 2001) with the addition of a list of entities, extracted from various knowledge bases, that describe persons, companies, and locations. We describe this process in detail in Section 5.

- **Structured-entity Identification.** A parallel rule-based task is used to extract entities that can easily be recognised without the need of training data. Among them: dates and times, numbers, email addresses, Italian "codice fiscale", that are based on textual patterns; roles and document types, that are based on prepacked lists.

- **Relation extraction (RE).** It is the task of extracting semantic relationships from text. Extracted relationships usually occur between two or more entities of a certain type (for example persons, locations, etc., see previous points), and fall into a number of semantic categories (such as birth location, role in a company, etc.). Relation extraction is widely used also in specific domains such as medicine (Giuliano et al., 2007) and finance (Vela and Declerck, 2009). Successful experiments made use of Conditional Random Fields (Surdeanu et al., 2011), Dependency-based Neural Networks (Liu et al., 2015), and transformers like BERT (Baldini Soares et al., 2019).

In this paper we present REDIT (Relation and Entities Dataset for Italian with Tint), a complete framework that aims to solve the personal data identification in textual documents. The software is mainly based on Tint (Palmero Aprosio and Moretti, 2018), an NLP pipeline specifically designed for Italian and based on Stanford CoreNLP (Manning et al., 2014). REDIT includes part of the annotated dataset (Section 4), the compiled model, and the supporting Java code. It is available for free on Github (see Section 6).

The content is structured as follows. Section 2 presents in detail how we collected the documents that are annotated and how we used fictitious data to make the resource available for download. In Section 3, we describe the process used to annotate the data. Section 4 illustrates the dataset, giving some statistics on the entities and relations included in it. In Section 5 we give some results on the performance of the resulting entity extraction and relation extraction system. The downloadable package (that contains the dataset, the model and the Java code) is finally described in Section 6.

2 Data Collection

The corpus is composed of documents taken from different institutions of the public administration. The documents with which we have worked are different types of forms, varying from license for parking to adoption forms, school enrollments, marriage licenses and so on.

Starting from this set, we create two datasets. One is composed of documents compiled with real data and one with documents compiled by us with fictitious data, using lists of all the Italian streets and surnames in order to guarantee the diversification of the data in the compiled forms, and to not exclusively rely on the annotators' fantasy. The fictitious compilation aims to avoid using sensitive data in terms of privacy issues, leading to the possibility of publicly releasing the dataset. The documents which contain real data are indeed not included in the public dataset. For instance, a sentence such as *Il sottoscritto Gianluca Freschi, nato a Pesaro il 12/12/1990 e residente in Pesaro, Via Virgilio n.76* presents data whose association was invented by the annotator. It could be possible that a person called *Gianluca Freschi* exists in real world but it is almost impossible that he would fit with the rest of the data since they all derive by annotator's fantasy. However, as we can see from the example, while the data are fictitious the structure of the document is identical to that of real ones.

3 The Annotation

Each document in the set is annotated both with entities and relations between them.

For the annotation of entities we adopt the guidelines already used for KIND (Paccosi and Palmero Aprosio, 2021), a corpus containing NE on documents taken from Wikinews. The named entities included in KIND belong to the standard

NE classes and are of three types: **LOC**, **PER**, and **ORG**. As already noticed by (Passaro et al., 2017), these categories are quite unsatisfactory to deal with the information contained in the PA documents, since the model is not designed at capturing information such as laws or protocols. In REDIT we then distinguish different types of ORGs, differentiating public offices and municipality and companies: the former is annotated as **ENTE**, while the latter as usual (**ORG**). Finally, we add a label to mark laws and protocols, **LEX**, so that in the present work there are five types of annotated entities: **LOC**, **PER**, **ORG**, **LEX**, and **ENTE**. The original guidelines used in KIND have therefore been slightly modified to meet our needs (see Section 5.1 for more details).

In addition to the NE annotation, we are interested in annotating the relations among them. In particular, we need to develop a system of relations which links the person with its personal data or with its role in terms of responsibility of the company/public administration or in terms of relative/family relationships.

Since for the annotation task a relation must connect two entities, some additional entity types are annotated only when involved in a relation (see below). The list of additional entities includes **ROLE** for personal and organisation roles (for example, words such as "responsabile", "titolare", "genitore", and so on, representing the role of a person in a company, in the PA domain, or in a family), **DOCTYPE** for document types (such as "passaporto", "patente"), **EMAIL** for e-mail addresses, **DATE** for dates, **NUMBER** for generic numbers (such as VAT), **CF** for the Italian "codice fiscale" sequence of chars.

Regarding relations, `address` is used for instance to link a LOC entity representing an address to the person or company to which the address belongs, while `birthDate`, `birthLoc` link respectively the date and location of birth.

Table 1 shows the complete list of the relations included in the dataset.

The annotation is performed by a domain expert using INCEpTION (Klie et al., 2018), a web-based text-annotation environment which allows users to: (i) select a group of tokens and assign a label to it (entities); (ii) connect two entities among them and assign a label to the link (relations).

This is an example of NER annotation:

> *Al [Comune di Alessandria]$_{ENTE}$.*
> *[Casale Monferrato]$_{LOC}$, 20 settembre 2021.*
> *Il sottoscritto [Davide Aiello]$_{PER}$, nato a [Milano]$_{LOC}$ il [31/07/1985]$_{DATE}$, [titolare]$_{ROLE}$ della ditta [Aiello Ceramiche S.r.l.]$_{ORG}$, ai sensi dell' [art. 76 del D.P.R. n. 445/2000]$_{LEX}$, dichiara di voler partecipare all'evento "Il mercante in Fiera".*

These are the corresponding relations:

- `birthLoc` (Davide Aiello, Milano)
- `birthDate` (Davide Aiello, 31/07/1985)
- `companyRole` (Davide Aiello, titolare)
- `personInOrg` (Davide Aiello, Aiello Ceramiche S.r.l.)

In the example, "31/07/1985" is tagged as DATE, since it is involved in the *birthDate* relation. On the contrary, since no relations include "20 settembre 2021", it's not mandatory, for the annotator, to mark it as DATE.

The system uses two different approaches to identify entities. Entities such as DATE or ROLE are annotated only when involved in a relation because they are labels identified through a rule-based approach which can be easily recognised without the need of training data. For what concerns instead PER, LOC, ORG, ENTE and LEX the identification occurs using a machine-learning technique and they need to be always annotated.

4 The Dataset

As we have seen in Section 2, the complete dataset consists of two parts: the first one presents the documents fictitiously compiled and it is publicly released; the latter, on the contrary, comprehends instances compiled with real data and is not released. Nevertheless, we consider also the unreleased dataset in training the model, so that the amount of annotated relations in the final dataset is 7,821, while that of annotated entities is 21,307. The released one presents 1,439 annotated entities and 1,476 annotated relations.

Looking at the data in Table 1, it is possible to notice that the amount of annotations referring to some relations (marked with *) are considerably fewer than others. Despite the small amount, we have already annotated them in the view of future works on these relations but we do not consider them in the experiments.

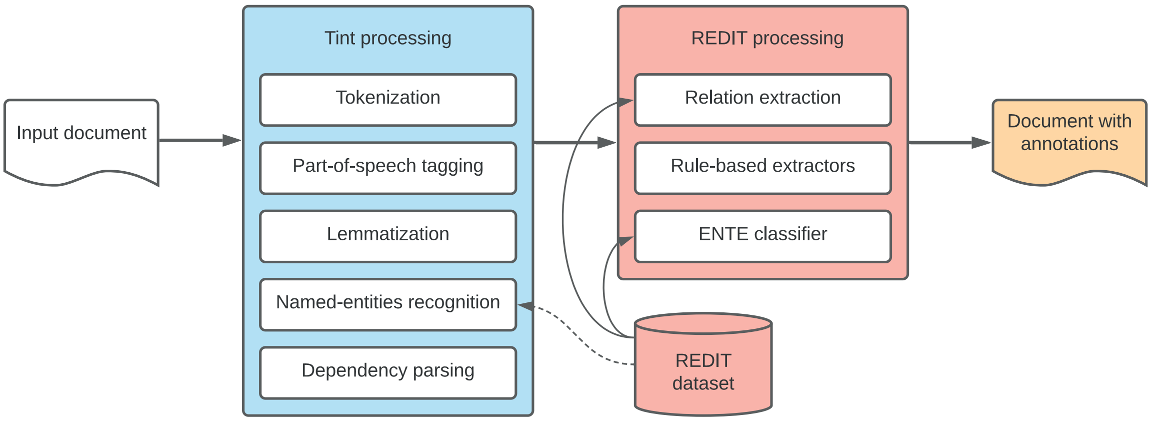


Figure 1: A chart depicting the REDIS architecture and its interaction with Tint.

Relation name	Released ds	Complete ds
address	451	2,115
birthDate	160	639
birthLoc	221	678
codiceFiscale	84	902
companyRole	59	99
deathDate (*)	5	6
deathLoc (*)	6	6
docExpDate (*)	2	2
docID (*)	13	13
docIssueDate (*)	8	8
docIssueLoc (*)	9	9
docType (*)	13	13
email	76	213
name (*)	28	28
personalRole	122	2,192
personInOrg	54	109
relative (*)	28	98
telephone	100	325
vat	37	366
Total	1,476	7,821

Table 1: Amount of annotated relations in the dataset.

Relation name	Released ds	Complete ds
ENTE	192	2829
LEX	214	8314
LOC	743	3788
ORG	62	2179
PER	228	4197
Total	1,439	21,307

Table 2: Amount of annotated entities in the dataset.

5 The Pipeline

To work properly, REDIT relies on a complex pipeline that includes various steps, very different in structure and management (see Figure 1). Most of the steps are performed using well-known tools and algorithms (sometimes not reaching state-of-the-art accuracy), so that the whole program does not need particular hardware (such as the GPUs needed in environments using deep learning and transformers) and is easy to run on almost every common software environment.

1. First, the input text is parsed with Tint (Palmero Aprosio and Moretti, 2018) using these annotators: tokenizer, sentence splitter, truecaser, part-of-speech tagger, lemmatizer, dependency parser.

2. Named-entities are extracted using the CRF implementation included in Stanford NER (Finkel et al., 2005) and the model trained on the annotated dataset (see Subsection 5.1).

3. A second run on named-entities, with the rule-based Stanford TokensRegex software (Chang and Manning, 2014), is performed (see Subsection 5.2)

4. ORG and LOC entities are passed into a Support Vector Machines classifier (Cortes and Vapnik, 1995) to extract ENTE entities (see Subsection 5.3).

5. Finally, the Stanford Relation Extractor (Surdeanu et al., 2011) is used to find relations between entities in the text (see Subsection 5.4).

Source	Tag	Labels
Wikipedia	LOC	377,611
Wikipedia	PER	608,547
Wikipedia	ORG	84,887
OpenStreetMap	LOC	389,649

Table 3: Items added to the NER training taken from gazettes.

Relation	P	R	F-score
LEX	0.762	0.760	0.761
LOC	0.830	0.811	0.820
ORG	0.832	0.821	0.826
PER	0.868	0.894	0.881
Total (micro)	**0.805**	**0.799**	**0.802**
Total (macro)	**0.823**	**0.821**	**0.822**

Table 4: Evaluation of the entity tagger.

5.1 The CRF Named-Entities Tagger

Since the sole REDIT dataset is not sufficient to train a robust NER tagger, we use it in combination with KIND (see Section 3). Guidelines for the two datasets are, of necessity, slightly different, therefore we need to use some precautions in merging them.

Sometimes, the entities annotated as ORG in KIND (such as "Unione Europea") should have been annotated as ENTE in REDIT. We then decided, in the training phase, to merge all ENTE entities into ORG. We then trained a classifier dedicated to the ENTE tag (see Subsection 5.3), trained on REDIT dataset only, that performs the sole disambiguation between ORG and ENTE.

To enhance the classification, Stanford NER also accepts gazettes of names labelled with the corresponding tag. We collect a list of persons, organizations and locations from the Italian Wikipedia using some classes in DBpedia (Auer et al., 2007): `Person`, `Organisation`, and `Place`, respectively. In addition to this, we collect the list of streets from OpenStreetMap (OpenStreetMap contributors, 2017), limiting the extraction to Italian names. Table 3 shows statistics about the gazettes.

The evaluation is performed by randomly splitting the dataset into train/dev/test using 80/10/10 ratio. During training phase, we tried some sets of features choosing among the ones available in Stanford NER. We obtained the best results (considering also a good balance between training/testing time and performances) with word shapes, n-grams with length 6, previous, current, and next token/lemma/class. Table 4 displays the results of the NER module.

5.2 The Rule-Based Named-Entities Tagger

As said in Section 3, there is the need for more entity types, because in the training phase we need to have both arguments of a relation annotates as an entity (of any type). For this reason, we use a rule-based approach to annotate DATE, ROLE, DOCTYPE, EMAIL, NUMBER, and CF.

- Tint TIMEX annotator is used to tag DATE entities.
- ROLE and DOCTYPE entities are extracted given a list of roles taken from the annotated training set.
- Numbers, e-mail addresses and Italian codice fiscale are tagged using regular expressions.

5.3 The SVM Classifier for ENTE Entities

After the previous steps, the entities that should be marked with ENTE now falls into the ORG or LOC entity sets. We then use a simple SVM classifiers (using shallow features, such as words, bigrams, previous and following content words, etc.) that, given an entity tagged as LOC or ORG, return whether it should be annotated as ENTE. The training set used by the classifier consists in entities taken from REDIT and annotated as ORG, LOC, and ENTE. The first two categories represent the zero class, while entities tagged with ENTE represent the other class. It is therefore a binary classifier. In a 10-fold cross-validation environment, results shows a F-score equals to 0.978 (precision 0.981, recall 0.974).

5.4 Relation Extractor Module

The last module in REDIT is Stanford Relation Extractor (Surdeanu et al., 2011), used to train and extract relations in the text.

Similarly to the NER training, we test approaches with different sets of features, obtaining the best results with unigrams/bigrams, adjacent words, argument words, argument class, dependency path between the arguments, entities and concatenation of POS tags between arguments.

Table 5 shows the results on the relation extractor (the evaluation is performed using gold-labeled entities).

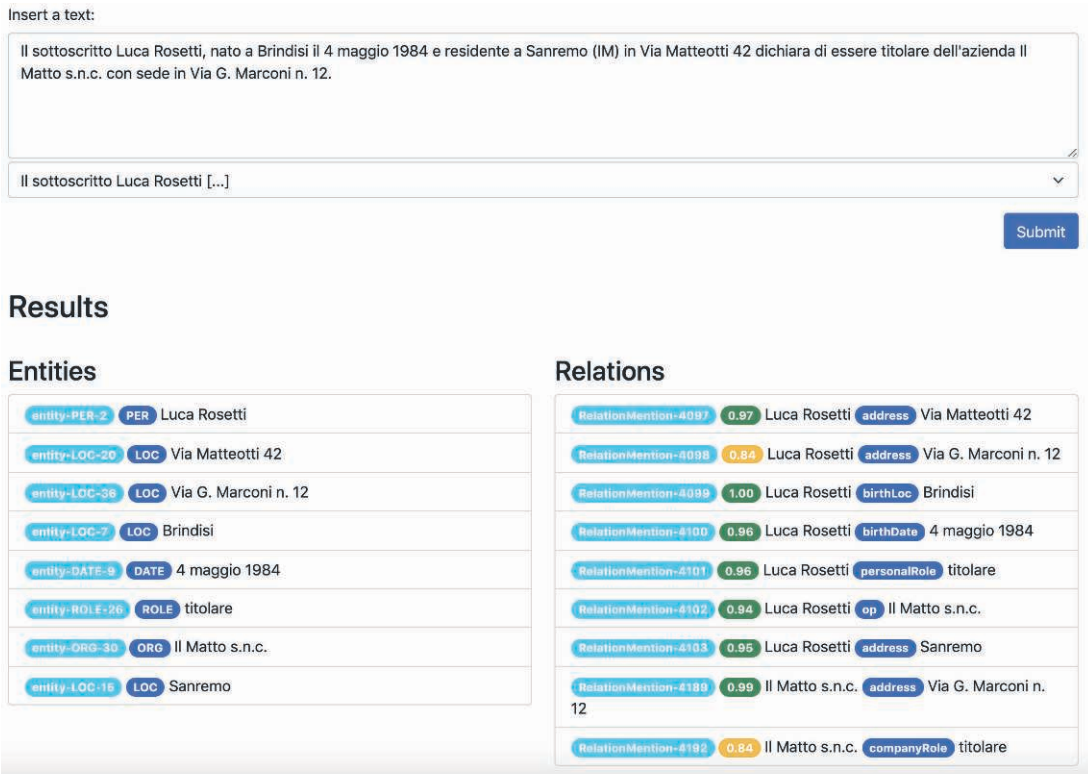


Figure 2: A screenshot of the demo interface.

Relation	P	R	F-score
address	0.929	0.908	0.918
birthDate	0.907	0.907	0.907
birthLoc	0.902	0.874	0.888
codiceFiscale	0.854	0.752	0.800
companyRole	0.902	0.676	0.773
email	0.865	0.421	0.566
personalRole	0.892	0.892	0.892
personInOrg	0.909	0.674	0.774
tel	0.935	0.580	0.716
vat	0.964	0.870	0.915
Total (micro)	**0.914**	**0.841**	**0.876**
Total (macro)	**0.906**	**0.755**	**0.824**

Table 5: Evaluation of the relation extractor.

6 The Release

All parts of REDIT (except part of the annotated dataset, see Section 4) are released for free under the CC BY 4.0 license,[2] and can be downloaded on Github.[3] These include the annotations, in WebAnno format (Yimam et al., 2013), the gazettes, both the NER and the RE models (created using the whole corpus), and the source code, written in Java, used to parse the files and run the classifiers.

A working demo of the tool is available online (See Figure 2).[4] Its web interface is written with VueJS/Boostrap and it is available for download in the Github project page.

7 Conclusion and Future Work

In this paper we present a completely automatic approach to extract personal data (view as entities) and relations between them from documents of the public administration written in Italian texts. The pipeline relies on a mix of rule-based and machine learning-base modules. The latter are trained using a manually annotated dataset, which is in part available for download. All the source code, instead, is released and available for download.

In the future, we plan to enhance the coverage of our system by adding more examples on relations that are less represented (see Table 1).

Acknowledgments

The research leading to this paper was partially supported by Wemapp Srl, Potenza, Italy.[5]

[2]https://bit.ly/cc-by-40-intl
[3]https://github.com/dhfbk/redit

[4]https://bit.ly/relation-extraction
[5]https://wemapp.eu/

References

Douglas Appelt, Jerry Hobbs, John Bear, David Israel, and Mabry Tyson. 1993. FASTUS: A Finite-state Processor for Information Extraction from Real-world Text. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1172–1178, 01.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In Karl Aberer, Key-Sun Choi, Natasha Noy, Dean Allemang, Kyung-Il Lee, Lyndon Nixon, Jennifer Golbeck, Peter Mika, Diana Maynard, Riichiro Mizoguchi, Guus Schreiber, and Philippe Cudré-Mauroux, editors, *The Semantic Web*, pages 722–735, Berlin, Heidelberg. Springer Berlin Heidelberg.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905, Florence, Italy, July. Association for Computational Linguistics.

I. Budi and S. Bressan. 2003. Association rules mining for name entity recognition. In *Proceedings of the Fourth International Conference on Web Information Systems Engineering, 2003. WISE 2003.*, pages 325–328.

Angel X. Chang and Christopher D. Manning. 2014. TokensRegex: Defining cascaded regular expressions over tokens. Technical Report CSTR 2014-02, Department of Computer Science, Stanford University.

Jason P. C. Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Claudio Giuliano, Alberto Lavelli, Daniele Pighin, and Lorenza Romano. 2007. FBK-IRST: Kernel methods for semantic relation extraction. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 141–144, Prague, Czech Republic, June. Association for Computational Linguistics.

Yongyan Guo, Jiayong Liu, Wenwu Tang, and Cheng Huang. 2021. Exsense: Extract sensitive information from unstructured data. *Comput. Secur.*, 102:102156.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9. Association for Computational Linguistics, Juni.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Yang Liu, Furu Wei, Sujian Li, Heng Ji, Ming Zhou, and Houfeng Wang. 2015. A dependency-based neural network for relation classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 285–290, Beijing, China, July. Association for Computational Linguistics.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

OpenStreetMap contributors. 2017. Planet dump retrieved from https://planet.osm.org . `https://www.openstreetmap.org`.

Teresa Paccosi and Alessio Palmero Aprosio. 2021. KIND: an Italian Multi-Domain Dataset for Named Entity Recognition. In *arXiv preprint*.

Alessio Palmero Aprosio and Giovanni Moretti. 2018. Tint 2.0: an all-inclusive suite for nlp in italian. In *Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it*, volume 10, page 12.

Emma Strubell, Patrick Verga, David Belanger, and Andrew McCallum. 2017. Fast and accurate entity recognition with iterated dilated convolutions.

Mihai Surdeanu, David McClosky, Mason Smith, Andrey Gusev, and Christopher Manning. 2011. Customizing an information extraction system to a new domain. In *Proceedings of the ACL 2011 Workshop on Relational Models of Semantics*, pages 2–10, Portland, Oregon, USA, June. Association for Computational Linguistics.

Mihaela Vela and Thierry Declerck. 2009. Concept and relation extraction in the finance domain. In H. Bunt, V. Petukhova, and S. Wubben, editors, *Proceedings of the Eighth International Conference on Computational Semantics (IWCS-8). International Conference on Computational Semantics (IWCS-8), January 7-9, Tilburg, Netherlands*, pages 346–351. Tilburg University, 1.

Seid Muhie Yimam, Iryna Gurevych, Richard Eckart de Castilho, and Chris Biemann. 2013. WebAnno: A flexible, web-based and visually supported system for distributed annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 1–6, Sofia, Bulgaria, August. Association for Computational Linguistics.

It Is MarkIT That Is New: An Italian Treebank of Marked Constructions

Teresa Paccosi, Alessio Palmero Aprosio, Sara Tonelli
Fondazione Bruno Kessler
Via Sommarive 18, Trento (Italy)
[tpaccosi|aprosio|satonelli]@fbk.eu

Abstract

English. In this paper we present MarkIT, a treebank of marked constructions in Italian, containing around 800 sentences with dependency annotation. We detail the process to extract the sentences and manually correct them. The resource covers seven types of marked constructions plus some ambiguous sentences, whose syntax can be wrongly classified as marked. We also present a preliminary evaluation of parsing performance, comparing a model trained on existing Italian treebanks with the model obtained by adding MarkIT to the training set.

Italiano. *In questo lavoro presentiamo MarkIT, un treebank di costruzioni marcate in italiano che contiene circa 800 frasi annotate con strutture a dipendenze. Abbiamo descritto nel dettaglio il processo seguito per estrarre le frasi e correggerne manualmente la struttura sintassi. La risorsa comprende sette tipologie di costruzioni marcate oltre ad alcune costruzioni ambigue che potrebbero essere classificate erroneamente come marcate. Presentiamo inoltre una valutazione preliminare delle performance del parser in cui confrontiamo un modello allenato sui treebank esistenti dell'italiano con il modello ottenuto aggiungendo anche MarkIT.*

1 Introduction

In recent years, the goal to develop robust frameworks for consistent annotation of syntactic dependencies across different human languages has led to the creation of Universal Dependencies (UD), an initiative covering nearly 200 treebanks in more than 100 languages. Since UD treebanks are then used to train syntactic parsers, it is important that they account for as many phenomena as possible that can be found in a language, and not only for canonical expressions typically written in news. The purpose to encompass the variety of use in the Italian language has been pursued by including different genres in the VIT treebank (Delmonte et al., 2007) and in ParTUT (Sanguinetti and Bosco, 2014) and more recently by including syntactically annotated tweets (Cignarella et al., 2019; Sanguinetti et al., 2018) in the UD framework. Overall, seven treebanks are listed under the UD initiative for Italian. In this work, we contribute to this effort by presenting a novel treebank including syntactically annotated marked constructions, which we call MarkIT (MARKed structures Italian Treebank). The samples have been extracted from a corpus of students' essays and to our knowledge represents the first effort to include in UD a repository of marked structures, which are typical of neo-standard language and are therefore more and more frequent in informal settings (D'Achille, 2003). The sentences have been first syntactically parsed and then manually corrected, so that we were also able to analyse which kinds of mistakes are typically done by dependency parsers. The dataset is freely available on Github at https://github.com/dhfbk/markit.

2 Related Work

In the last years, Universal Depedencies (UD) have become the most widely used standard for syntactic annotation (de Marneffe and Manning, 2008) upon which treebanks for other languages have been built, including Italian. The first one has been the Italian Stanford Dependency Treebank or ISDT (Bosco et al., 2013). Other treebanks have been later built with different purposes, covering a rich collection of different usages and genres. In particular, the VIT treebank (Delmonte et al., 2007) is composed of several texts ranging from news to literature, while TWITTIRO (Cignarella et al., 2019) and PoSTWITA (Sanguinetti et al., 2018) are two social-media-based treebanks, composed of tweets. These two Twitter-based treebanks represent an important resource in terms of documentation of the usage of non-standard Italian. We address the same topic in the present work, but instead of considering social-media data, we look at more formal writings, and in particular at the use of marked sentence constructions in students' essays. To our knowledge, a grammatical UD treebank for Italian language does not exist, and also in other languages there are only few examples. A grammatical treebank is a dataset of annotated trees sharing the same type of grammatical constructions, such as the English Pronouns treebank (Munro, 2021), which is the most similar resource to ours. It was created to make independent genitive pronoun's identification more accurate, by annotating only English sentences which display that construction. For what concerns marked structures in Italian, a comparative study on the distribution of the phenomenon of syntactic markedness has been presented in (Pieri et al., 2016), but the different structures were identified using automated tools. Overall, syntactic markedness is a phenomenon poorly analyzed, especially in the field of dependency grammar. However, it is crucial to make parsers more robust to different syntactic structures.

3 Sentence Collection

Our goal is to build a treebank of marked constructions that reflects actual usage of Italian, in particular of the neo-standard variant (Berruto, 2012). We avoid to manually create sentences ourselves, also to increase linguistic variability. Therefore, we resort to a corpus of students' essays which were collected by Istituto provinciale Trentino per la Ricerca e la Sperimentazione educativa (IPRASE) with the goal to study the evolution of high-school students' writing skills, taking into account essays spanning 15 years (from 2001 to 2016). In particular, the project tracked the presence of expressions and constructions typical of neo-standard Italian, requiring a pool of expert annotators, i.e. high-school teachers, to manually mark in essays a number of linguistic traits (Sprugnoli et al., 2018; Tonelli et al., 2020). Among others, annotators were asked to mark dislocated sentences, cleft sentences and hanging topics (see details in Section 4). These were first automatically identified through the TINT NLP Suite (Aprosio and Moretti, 2018) and then manually revised by annotators to distinguish between the constructions of interest and other types of similar constructions.

The final corpus contains more than 2,500 essays and almost 1.5 million tokens. We extract around 800 sentences labeled with a marked structure and annotate them at syntactic level. Although the essays cannot be released because of copyright issues, the sentences in isolation, with no additional information related to the authors or the textual context, can be freely distributed.

The essays were written in a time span of 15 years by different authors and dealing with a number of different topics, which guarantees a high variability of the sentence content and structure. On the other hand, since they were part of a formal students' examination, they tend to be free from jargon, grammatical errors and abbreviations that may derive from sentences extracted from social media and that may represent an additional challenge for parsers.

4 Marked Structures in IPRASE Corpus

With marked sentences we refer to those constructions which present a non-canonical order of constituents. In Italian, the canonical order of the syntactic structure is *S V+fin V-fin OX*, where *S* is subject, *V+fin* is a finite verb or an auxiliary verb, *V-fin* is a non-finite verb, *O* is the direct object and *X* other complements (Benincà et al., 1988). Marked structures are instead intended to focus on an element of the sentence, by moving the focalized constituent in a different position from the one it occupies in a canonical sentence. The reason for markedness in Italian can be phonotactic or bound to the whole meaning of the sentence. In syntactical terms, we can say that the marked structures operate a modification in the distribution of *topic* and *comment* with respect to the corresponding non marked structure (Cinque, 1990). There are seven possible marked structures in Italian: sentences with postverbal subject, sentences with presentative "there", sentences with left or right dislocation, hanging topic sentences, cleft sentences

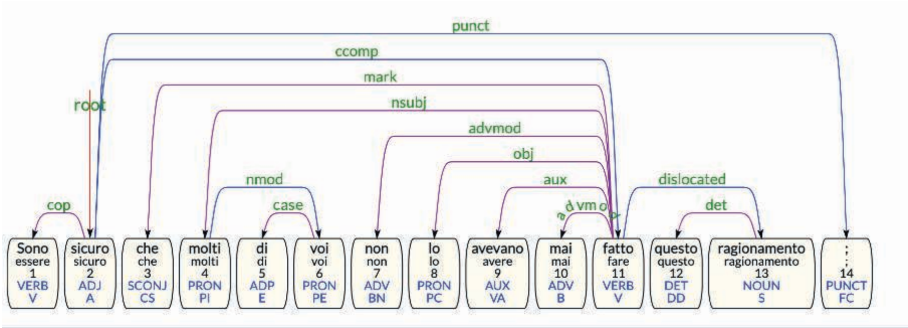


Figure 1: Right dislocated sentence annotated with dislocated relation

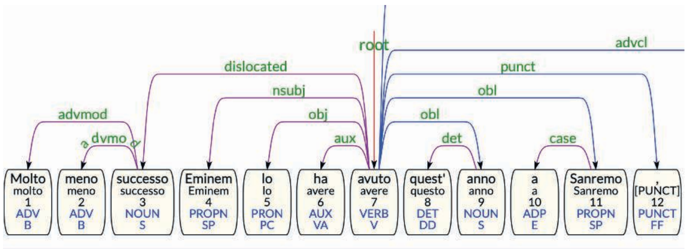


Figure 2: Left dislocated sentence annotated with dislocated relation

and pseudo-cleft sentences (Ferrari and Zampese, 2016). Among the sentences from the IPRASE corpus originally marked as dislocated, cleft and hanging topic, we were able to find other types of marked structures which had been wrongly identified by the annotators, so that in the end all seven phenomena are present. Below we report a brief description of the main marked structures annotated in our treebank.

4.1 Left Dislocated Sentences

Left dislocated sentences entail the displacement or anteposition of a specific syntagm to the left of the sentence. The dislocated element connects with the rest of the sentence thanks to an introductory preposition (1) or a pronominal reprise (2), for which a resumptive clitic pronoun pleonastically co-refers to the displaced nominal element (the topic). The clitic reprise is compulsory whether the displaced element was the direct object, as long as it is in the positive form (Benincà et al., 1988).

(1) A questo evento (ci) partecipano soltanto artisti già noti
To this event (clitic) participate only artists already known

(2) Molto meno successo Eminem lo ha avuto quest'anno
Much less success Eminem it has had this year

4.2 Hanging Topic Sentences

In hanging topic sentences, similarly to left dislocation, the dislocated element is moved to the left, at the beginning of the sentence. However, in this case, the displaced element is isolated at the beginning of the sentence, and it is not syntactically linked to the verb (D'Achille, 2003). The main difference between the two structures is when the dislocated element is the direct object. In fact, since direct objects in Italian exclude prepositional government, only the non-clitic reprise allows the distinction between left dislocated sentences and hanging topics. In hanging topic constructions, the isolated element is always deprived of indicators for its syntactic function, and it is typically reprised in the following phrase by different anaphorical expressions such as atonic pronouns, possessive pronouns, adverbs, and by a whole nominal phrase (3). When there is no reprise of the dislocated element in the subsequent sentence, we refer to that as an example of anacoluthon (Ferrari and Zampese, 2016).

(3) [...] ma il cervello, senza di esso non siamo niente
But the brain, without it we are nothing

4.3 Right Dislocated Sentences

Right dislocated sentences operate a topicalization of the comment and, differently from left dislocated structures, the pronominal reprise is not

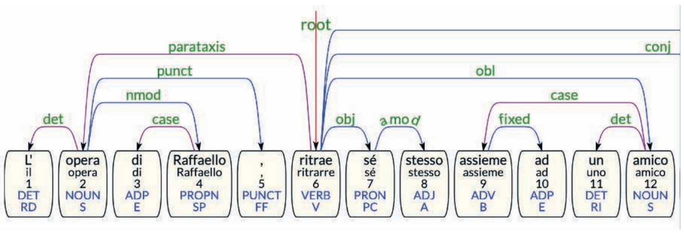


Figure 3: Hanging topic annotated with parataxis relation

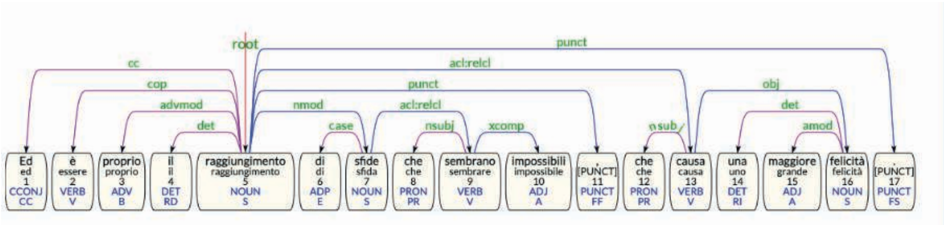


Figure 4: Cleft sentence with relative clause (*acl:relcl*)

compulsory when the dislocated element is the direct object. Nevertheless, since the non-marked position of the right dislocated elements is still in postverbal position (apart from the subject), it makes the presence of the anticipatory clitic (4) or of the comma (5) compulsory.

(4) Sono sicuro che molti di voi non lo avevano mai fatto, questo ragionamento
I am sure that many of you do not it have never done, this reasoning

(5) Interessante, in questo senso, la riflessione di Paul
Interesting, in this sense, Paul's thought

4.4 Cleft Sentences

Cleft sentences are typically composed of a main clause without a subject introduced by the verb 'to be' in different forms, followed by the cleft constituent and by a subordinate clause introduced by "che" (*that*), whose function can be of relative pronoun (6) or relative conjunction. Sometimes, the subordinate clause can be introduced by "a" (*to*) + a verb in the infinite form (7), if the subject is the element to put into focus (Berruto and Cerruti, 2011). Besides the subject, cleft structures can focalize on several constituents, such as the object, prepositional constituents, adverbs and also verbs, especially in the infinitive form (Renzi, 2001).

(6) È lo Stato che [...] impone i suoi modelli
It is the State that [. . .] imposes its models

(7) Non è dunque l'ottica dell'utilità e del guadagno a guidare verso la felicità
It is not then the view of utility and profit to guide to happiness

5 MarkIT Annotation

Marked structures, such as the ones described above, are very difficult to parse, since they belong to non-standard Italian constructions. In order to annotate them syntactically, we therefore need to follow a semi-automatic approach, by analysing them first with a dependency parser and then manually correcting them. The selected marked constructions from the IPRASE corpus were processed with the TINT parsing module (Aprosio and Moretti, 2018), which is built following Universal Dependencies guidelines (de Marneffe and Manning, 2008), and trained on the Italian Stanford Dependency Treebank, ISDT (Bosco et al., 2013). ISDT includes mostly standard language with few non-canonical constructions. The dependency trees parsed by TINT are then manually corrected by an expert linguist using the TINTful interface (Frasnelli et al., 2021). They are also marked with one of the categories from Table 1.

Concerning *dislocated sentences*, the main issue with TINT is that it assigns to the pronoun the role of direct object and treats the dislocated element as the subject, as in the example shown in Figure 6. The sentence was manually corrected by marking the dislocated element with the *dislocated* relation and the pronoun of reprise with the core argument relation which it represents (*obj* or *subj*), as we can see in Fig. 1 and Fig. 2.

As previously mentioned, *hanging topics* differ from left dislocated sentences because the element to the left is not syntactically linked to the

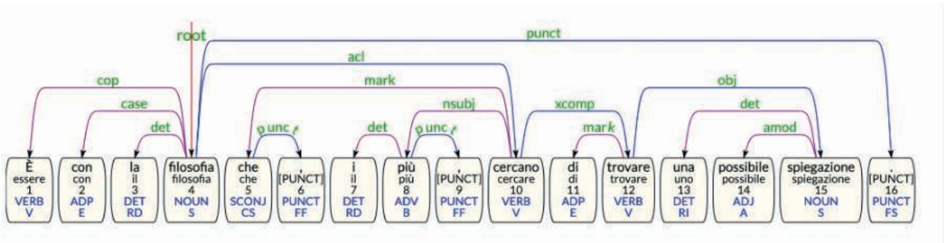


Figure 5: Cleft sentence with adnominal clause (*acl*)

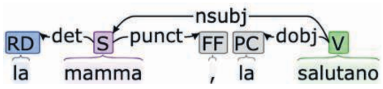


Figure 6: Wrong parsing output of dislocation

verb and there is no clitic reprise of the lexical element. Since there is a sort of isolation of the topicalized element, we choose to use the *parataxis* relation to link it to the head of the sentence, since *parataxis* is defined as a relation between a word and other elements, without any explicit coordination, subordination, or argument relation with the head word, which is usually the verb. An example is reported in Figure 3.

As we have seen above, the "che" (*that*) before a subordinate clause can be a relative pronoun introducing a relative clause or a relative conjunction, followed by a structure whose nature is controversial. A relative clause is an instance of clausal modifier *acl*, which takes the specific name of *acl:relcl*, where the noun can be omitted or substituted by a relative pronoun, relative conjunction, or an adverb. As regards *cleft sentences*, we choose to use the same relation in two different ways, in order to distinguish between the case in which the cleft sentence comprehends a relative clause or an unspecified subordinate clause. When the dislocated element is the subject or the direct object (substituted by a relative pronoun) we use the *acl:relcl* relation, selecting the role of "che" (see Fig. 4). Instead, if there is no dislocation of the subject or the object, we use the *acl* relation but we do not select the function of "che". Indeed, "che" is treated as a mere introducer for the subordinate clause with the *mark* relation (Fig. 5).

Table 1 shows the eight types of constructions in MarkIT. Beside the four types of marked structures described above, we found several other structures coming from the erroneous identification of cleft sentences and right dislocated sentences in the original IPRASE corpus. "Presentative there" (i.e. "there" + verb to be + nominal element + that)[1] and "pseudo-cleft" structures ("what" clause + verb to be) [2] were wrongly identified as cleft sentences, while the structures with a postverbal subject were originally labeled as right dislocated. Furthermore, we include in the "Other" category the structures which resulted challenging to tag for the annotators and which are not "presentative there" nor pseudo-cleft constructions. "Other" structures are namely those which usually present an explicit subject in the main clause and are erroneously identified as cleft, for example *La capacità di concepire un insieme di diritti è una facoltà che distingue l'uomo dagli altri esseri viventi* (EN: The ability to conceive a set of rights is a faculty that distinguishes humans from other living beings). "Other" structures include also passive clauses, which were originally tagged as right dislocated because of the postverbal position of the subject. Sentences in this last category are particularly challenging both for parsers and for human annotators, since they were wrongly classified even by IPRASE experts (i.e. high-school teachers) and have been assigned the correct label only after our revision.

6 Parsing Evaluation

As already mentioned in the Introduction, the lack of marked structures in treebanks used to train syntactic parsers may affect the system robustness, since structures which are not represented in the training data tend to be poorly analysed. In order to measure the impact of our novel treebank on the dependency analysis of marked structures, we compare the performance of the parser included in TINT, part of Stanford CoreNLP (Manning et al., 2014) by testing it on the new annotated sentences and training on different datasets. In particular,

[1] e.g. *C'è Michela che ti cerca* (EN: There is Michela that is looking for you)

[2] e.g. *Ciò che voglio davvero è che tu te ne vada* (EN: What I really want is that you go)

Type	Sents
Cleft sentences	309
Left dislocated	121
Right dislocated	49
Presentative "there"	25
Postverbal subject	16
Pseudo-clefts	11
Hanging topic	7
Other	275
Total	813
Total (tokens)	24,623

Table 1: Number of examples in the dataset.

we first split our novel treebank into training, dev, and test, respectively 80%, 10%, and 10%, proportionally with respect to the categories listed in Table 1. When the number of examples is tiny, we include a minimum of two examples for each class in each split, therefore test and dev set contain two examples of *hanging topic* each, leaving the three sentences for the training set.

We then compare two models: the original neural transition-based parser model used by TINT, which is trained using ISDT, VIT, and ParTUT (see Section 2), and the model obtained by adding to the above training data also the training set of MarkIT. We choose not to include the other Italian datasets available from Universal Dependencies (such as the ones derived from Twitter) because of their particularly informal language, which is very different from MarkIT sentences taken from students' essays. In both cases, we use the concatenation of the development sets of the four datasets as development set during the training phase. Following the standard evaluation used in dependency parsing, we compute unlabeled attachment score (UAS) and labeled attachment score (LAS) in the two tests.

Training set	UAS	LAS
ISDT+VIT+ParTut	82.53	76.62
ISDT+VIT+ParTut+MarkIT	82.74	77.41

Table 2: Evaluation of the dependency parsing.

Results in Table 2 show that on the one hand adding MarkIT to the training set improves the classification of marked structures, but on the other hand performance gain is limited. This may be due to the fact that, compared to the other treebanks (more than 23k sentences in total), the number of training instances coming from MarkIT is small (around 650 sentences). More generally, the presence of both marked and not marked sentences (the "Other" category) in the test set represents a challenge for parsers, since very similar constructions are labeled differently, see for example the presence of comma to mark right dislocated elements. Indeed, if the first model is tested only on sentences taken from ISDT+VIT, it achieves 84.47 UAS and 80.69 LAS.

7 Release

MarkIT is released under CC BY 4.0 license,[3] and can be downloaded from Github.[4] The annotation of the treebank will be soon completed with all marked sentences in the essays dataset (see Section 8) and proposed for publication on the Universal Dependencies website.[5] Since the treebank is still being extended with new sentences, it may be that the content of the last version available online exceeds the size of the resource described in this paper.

8 Conclusions

In this work we present MarkIT, a novel treebank composed of 800 sentences with syntactic annotation of marked structures. The resource covers seven types of marked sentences, plus around 200 sentences whose structure is not marked but that may be misleading both for parsers and for human annotators. The treebank is made available to the community and is meant to make dependency parsers more robust to the different syntactic structures present in Italian, in particular in the neo-standard variant. The work is still in progress, since we plan to add to the resource other sentences from the IPRASE corpus. Our goal is to include all marked sentences present in the essays, so to analyse also the distribution of the different sentence structures in this type of texts.

References

Alessio Palmero Aprosio and Giovanni Moretti. 2018. Tint 2.0: an all-inclusive suite for nlp in italian. In

[3] https://bit.ly/cc-by-40-intl
[4] https://github.com/dhfbk/markit
[5] https://universaldependencies.org/

Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it, volume 10, page 12.

P. Benincà, Salvi G., and Frison L. 1988. L'ordine degli elementi della frase e le costruzioni marcate. In L. Renzi, editor, *Grande grammatica italiana di consultazione. I. La frase. I sintagmi nominale e preposizionale*, pages 115–225. Il Mulino.

G. Berruto and M. Cerruti. 2011. *La linguistica: un corso introduttivo*. UTET Università.

Gateano Berruto. 2012. *Sociolinguistica dell'italiano contemporaneo*. Carocci.

Cristina Bosco, Simonetta Montemagni, and Maria Simi. 2013. Converting Italian treebanks: Towards an Italian Stanford dependency treebank. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 61–69, Sofia, Bulgaria, August. Association for Computational Linguistics.

Alessandra Teresa Cignarella, Cristina Bosco, and Paolo Rosso. 2019. Presenting TWITTIRÒ-UD: An Italian Twitter treebank in Universal Dependencies. In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*, pages 190–197, Paris, France, August. Association for Computational Linguistics.

Guglielmo Cinque. 1990. *Types of A' Dependencies*. MIT Press.

Paolo D'Achille. 2003. *L'italiano contemporaneo*. Il mulino Bologna.

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The Stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8, Manchester, UK, August. Coling 2008 Organizing Committee.

R Delmonte, A Bristot, and Sara Tonelli. 2007. Vit-venice italian treebank: Syntactic and quantitative features. In *Sixth International Workshop on Treebanks and Linguistic Theories*, volume 1, pages 43–54. Northern European Association for Language Technol.

A. Ferrari and L. Zampese. 2016. *Grammatica: parole, frasi, testi dell'italiano*. Carocci editore.

Valentino Frasnelli, Lorenzo Bocchi, and Alessio Palmero Aprosio. 2021. Erase and rewind: Manual correction of NLP output through a web interface. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 107–113, Online, August. Association for Computational Linguistics.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Robert M. Munro. 2021. *Human-in-the-Loop Machine Learning: Active learning and annotation for human-centered AI*. Simon and Schuster.

Giulia Pieri, Dominique Brunato, and Felice Dell'Orletta. 2016. Studio sull'ordine dei costituenti nel confronto tra generi e complessità (analysis of constituents order across textual genres and complexity). In Pierpaolo Basile, Anna Corazza, Francesco Cutugno, Simonetta Montemagni, Malvina Nissim, Viviana Patti, Giovanni Semeraro, and Rachele Sprugnoli, editors, *Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016), Napoli, Italy, December 5-7, 2016*, volume 1749 of *CEUR Workshop Proceedings*. CEUR-WS.org.

L. Renzi. 2001. Le tendenze dell'italiano contemporaneo. note sul cambiamento linguistico nel breve periodo. *Studi di lessicografia italiana*, pages 279–319.

Manuela Sanguinetti and Cristina Bosco. 2014. Converting the parallel treebank partut in universal stanford dependencies. *Converting the parallel treebank ParTUT in Universal Stanford Dependencies*, pages 316–321.

Manuela Sanguinetti, Fabio Poletto, Cristina Bosco, Viviana Patti, and Marco Stranisci. 2018. An Italian Twitter corpus of hate speech against immigrants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Rachele Sprugnoli, Tonelli Sara, Alessio Palmero Aprosio, and Moretti Giovanni. 2018. Analysing the evolution of students' writing skills and the impact of neo-standard italian with the help of computational linguistics. In *CLiC-it 2018 Italian Conference on Computational Linguistics*, pages 354–359. aAccademia University Press.

Sara Tonelli, Rachele Sprugnoli, Alessio Palmero Aprosio, Moretti Giovanni, and Menini Stefano. 2020. Gli strumenti informatici. sviluppo e risultati. In Michele Ruele and Elvira Zuin, editors, *Come cambia la scrittura a scuola: Rapporto di ricerca*, chapter 4, pages 113–130. IPRASE.

Linguistic Cues of Deception in a Multilingual April Fools' Day Context

Katerina Papantoniou[1,2], **Panagiotis Papadakos**[2], **Giorgos Flouris**[2], **Dimitris Plexousakis**[1,2]
1. Computer Science Department, University of Crete, Greece
2. Institute of Computer Science, FORTH, Greece
{papanton, papadako, fgeo, dp}@ics.forth.gr

Abstract

In this work we consider the collection of deceptive April Fools' Day (AFD) news articles as a useful addition in existing datasets for deception detection tasks. Such collections have an established ground truth and are relatively easy to construct across languages. As a result, we introduce a corpus that includes diachronic AFD and normal articles from Greek newspapers and news websites. On top of that, we build a rich linguistic feature set, and analyze and compare its deception cues with the only AFD collection currently available, which is in English. Following a current research thread, we also discuss the individualism/collectivism dimension in deception with respect to these two datasets. Lastly, we build classifiers by testing various monolingual and crosslingual settings. The results showcase that AFD datasets can be helpful in deception detection studies, and are in alignment with the observations of other deception detection works.

1 Introduction

April Fools' Day (for short AFD) is a long standing custom, mostly in Western societies. It is the only day of the year when practical jokes and deception are expected. This is the case for all social interactions, including journalism, which is generally considered to aim at the presentation of truth. Every year on this day, newspapers and news websites take part in an unofficial competition to invent the most believable, but untrue story. In this respect, AFD news articles fall into the deception spectrum, as they satisfy widely acceptable definitions of deception as in Masip et al. (2005).

The massive participation of news media in this custom establishes a rich corpus of deceptive articles from a diversity of sources. Although AFD articles may exploit common linguistic instruments with satire news, like exaggeration, humour, irony and paralogism, they are usually considered a distinct category. This is mainly due to the fact that they also employ other mechanisms which characterize deception in general, like sophisms, and changes in cognitive load and emotions (Hauch et al., 2015) to deceive their audience. AFD articles are often believable, and there exist cases where sophisticated AFD articles have been reproduced by major international news agencies worldwide[1].

This motivated us to extend our previous work on linguistic cues of deception and their relation to the cultural dimension of individualism and collectivism (Papantoniou et al., 2021), in the context of the AFD. That work examines if differences in the usage of linguistic cues of deception (e.g., pronouns) across cultures can be identified and attributed to the individualism/collectivism divide.

Specifically, the contributions of this work are:

- A new corpus that includes diachronic AFD and normal articles from Greek newspapers and news websites[2], adding one more AFD collection to the currently unique one in English (Dearden and Baron, 2019).
- A study and discussion of the linguistic cues of deception that prevail in the Greek and English collection, along with their similarities.
- A discussion on whether the consideration of the individualism/collectivism cultural di-

[1] https://www.nationalgeographic.com/history/article/150331-april-fools-day-hoax-prank-history-holiday

[2] The collection is available in: https://gitlab.isl.ics.forth.gr/papanton/elaprilfoolcorpus

mension in the context of AFD aligns with the results of our previous work.

- An examination of the performance of various classifiers in identifying AFD articles, including multilanguage setups.

2 Related Work

The creation of reliable and realistic ground truth datasets for the deception detection task is a challenging task (Fitzpatrick and Bachenko, 2012). Crowdsourcing, in the form of online campaigns in which people express themselves in truthful and/or deceitful manner for a small payment are a well established way to collect deceptive data (Ott et al., 2011). Real-life situations such as trials (Soldner et al., 2019) or the use of data from board games have also been employed (Peskov et al., 2020). Also a popular approach is the reuse of content from sites that debunk articles like fake news and hoaxes (Wang, 2017; Kochkina et al., 2018). Lastly, satire news are another way to collect deceptive texts, but with some particularities due to humorous deception (Skalicky et al., 2020).

The only work that explores AFD articles is that of Dearden et al. (2019). They collected 519 AFD and 519 truthful stories and articles in English for a period of 14 years. A large set of features was exploited to identify deception cues in AFD stories. Structural complexity and level of detail were among the most valuable features while the exploitation of the same feature set to a fake news dataset resulted in similar observations.

To the best of our knowledge, the only deception related dataset for the Greek language is that of Karidi et al. (2019). This work proposed an automatic process for the creation of a fake news and hoaxes articles corpus, but unfortunately the created corpus over Greek websites is not available. If we also consider that the creation of a Greek dataset for deception through crowdsourcing is a cumbersome and expensive task, that is further hindered by the exceptionally limited number of native Greek crowd workers, it is easy to understand why there is a lack of datasets.

Regarding the individualism/collectivism cultural dimension, it constitutes a well-known division of cultures that concerns the degree in which members of a culture value more individual over group goals and vice versa. In individualism, ties between individuals are loose and individuals are expected to take care of only themselves and their immediate families, whereas in collectivism ties in society are stronger. In Papantoniou et al. (2021) there is an preliminary effort driven by prior work in psychology discipline (Taylor et al., 2017) to examine if deception cues are altered across cultures and if this can be attributed to this divide. Among the conclusions were that people from individualistic cultures employ more third and less first person pronouns to distance themselves from the deceit when they are deceptive, whereas in the collectivism group this trend is milder, signalling the effort of the deceiver to distance the group from the deceit. In addition, in individualistic cultures positive sentiment is employed in deceptive language, whereas in collectivists there is a restraint of expression of sentiment both in truthful and deceptive texts.

To this end, this work explores the deception-related characteristics of a new Greek corpus based on AFD articles from a variety of sources, and compares them with the English ones[3]. Further, since related studies (Triandis and Vassiliou, 1972; Hofstede, 1980; Koutsantoni, 2005) describe Greece as a culture with more collectivistic characteristics (by using country as proxy from culture), we also discuss differences in deception cues along this cultural dimension.

3 Corpus Creation

The AFD articles have been hand gathered because a crawling based collection approach was not applicable in our case. Since the news web sites industry in Greece is not huge to establish an acceptable number of crawled AFD articles, we had to additionally collect articles from the press, including articles from the pre-WWW era. Specifically, we visited the local library that maintains a printed archive of newspapers and searched for disclosure articles in the issues after the 1st April, took photos of the AFD articles, and then used OCR and manual inspection to extract the text. In addition we contacted national and local news media providers to get access in their digitalized archives. The rest were gathered from the Web.

The articles were categorized thematically into the following five categories: society, culture, politics, world, and sports. If no category was pro-

[3]We also experimented with data from the limited number of satirical and hoaxes sources of the Greek Web. We do not discuss them here though, since the classifiers reported excellent accuracy showcasing the lack of diversity and the existence of domain specific information in the collected data.

vided by the original source, we manually annotated the articles. For each article we kept the title, the main body, the published date, the name, the type of the source (newspaper or news website), and (if available) the caption, the subtitle and the author. As preprocesing steps we applied spellcheck and normalization. The correction of spelling mistakes was necessary primarily for articles extracted through OCR tools, although spelling errors were identified in other articles too. Normalization was performed for homogeneity reasons in the texts retrieved from the 80's, since we observed language differences in some forms (e.g., in the suffix of genitive case), which are remains of an old form of Modern Greek[4].

For the truthful collection we used the same manual procedure and we tried to have a balanced dataset in terms of thematic categories. The truthful collection consists of articles that have been published in days relatively close to the 1st of April in order to have articles that do not differ significantly in respect to their topics, mentioned named entities, etc.

Since the AFD tradition is vivid in Greece, we were able to locate a lot of such articles from various newspapers and new websites for our corpus (112 different sources). Specifically, we managed to collect 254 truthful and 254 deceptive articles spanning over the period 1979 - 2021. In Tables 1 to 2 some statistics of the corpus are depicted.

Measure	Truthful	Deceptive
Num. of articles	254	254
Avg. length	336	255
Min. length	57	33
Max. length	1347	1163

Table 1: Overview of the dataset.

Topic	Truthful	Deceptive
culture	20	24
politics	85	78
society	86	118
sports	22	29
world	41	5

Table 2: Distribution of articles per topic.

[4]https://en.wikipedia.org/wiki/Katharevousa

4 Features Analysis

For the analysis of AFD articles we adapt and build upon the feature set used in Papantoniou et al. (2021), but for the Greek language. The resulting feature set consists of 64 features for the Greek language and 75 for the English, due to the smaller availability of linguistic resources for Greek (e.g., in sentiment lexicons). For the analysis we performed the non-parametric Mann–Whitney U test (two-tailed) with a 99% confidence interval (CI) and $\alpha = 0.01$. Table 3 depicts the results of this analysis for elAFD and enAFD datasets[5].

In both datasets, positive sentiment is related to the deceptive articles, while negative sentiment with the truthful articles. The only exception concerns the enAFD dataset, where for the NRC lexicon the opposite holds (NRC is one of the six sentiment lexicons used for features in English). In addition, negative emotions like anger, fear and sadness are related to truthful news articles in both datasets. The use of positive emotive language during deception may be a strategy for deceivers to maintain social harmony as noticed also by other studies (Newman et al., 2003; Pérez-Rosas et al., 2018). The difference in the use of emotional language between truthful and deceptive news is more intense in the enAFD dataset, where five out of the eight emotions in the NRC lexicon are found statistical significant. This is in alignment with the results in Papantoniou et al. (2021) for individualistic and collectivistic cultures.

Further, deceptive texts seem to be related with an increased use of adverbs in both datasets. This can be related to the less concreteness of deceptive texts as discussed in Kleinberg et al. (2019) and it is in line with many theories of deception like the Reality Monitoring (Johnson et al., 1998), Criteria based Content Analysis (Undeutsch, 1989) and Verifiability Approach (Nahari et al., 2014). This also explains the prevalence of the number of named entities, spatial related words, conjunctions and WDAL imagery score in truthful texts in the enAFD dataset and the use of more motion verbs in deceptive texts in the elAFD dataset. According to cognitive load theory (Sweller, 2011) in deceptive texts the language is less specific and consists of simpler constructs. The same holds for modality, another common feature among the datasets, that is considered a signal of subjectivity that pro-

[5]All the features are described in
https://gitlab.isl.ics.forth.gr/papanton/elaprilfoolcorpus

vides a degree of uncertainty. In addition, hedges in enAFD dataset, also express some feeling of doubt or hesitancy.

Lexical diversity as expressed by the token-type ratio (TTR), that is the ratio of unique words to the total number of tokens, is related to the deceptive texts. This seems to contradict all the above, but could be attributed to the fact that deceptive texts are shorter. Although this is more evident in the case of the enAFD dataset, it also holds for elAFD dataset (see Table 1).

Boosters, which are words that express confidence (e.g., certainly) are quite discriminative for deceptive texts for the enAFD dataset. Moreover we observe the connection of the future tense with deception and of the past with truth. The above were also marked in Papantoniou et al. (2021) in different domain from the news articles domain.

Finally, first personal pronouns have been found to be rather discriminative of deceptive texts in various deception detection and cultural studies, including Papantoniou et al. (2021). However, in this study pronouns are statistical important only for the enAFD dataset. This probably reflects idiosyncrasies of the news domain, since articles mainly present objectively facts and not opinions, and as a result the use of first personal pronouns is avoided. This holds for the elAFD dataset that includes AFD articles from the news sites and the press, and not for the enAFD dataset that consists of various types of AFD articles and stories collected from the web through crowdsourcing[6].

Deceptive	Truthful
elAFD	
adverbs (0.31)	punctuation (-0.17)
adj. & adv. (0.27)	nrc sadness(-0.17)
TTR (0.27)	plosives (-0.16)
pos. sentiment (0.21)	nrc anger (-0.15)
modal verbs (0.17)	nrc fear (-0.14)
motion verbs (0.117)	vowels (-0.14)
	consonants (-0.14)
enAFD	
boosters (0.39)	NE num. (-0.27)
modal verbs (0.35)	spatial num. (-0.26)
TTR(0.31)	conjuctions (-0.24)
future (0.27)	nrc fear (-0.23)
adverbs (0.2)	past (-0.23)
1st pers. pp (0.2)	nrc sadness (-0.23)
mpqa pos. (0.2)	nrc anger (-0.21)
nrc neg.* (-0.2)	nrc trust (-0.21)
2nd pers. pp (0.19)	avg. word len. (-0.17)
1st pers. pp pl. (0.18)	collectivism (-0.16)
sentiwordnet pos. (0.17)	nrc pos.* (-0.16)
demonstrative (0.17)	wdal imagery (-0.15)
hedges (0.17)	mpqa neg. -0.14)
adj & adv (0.16)	nasals (-0.14)
present (0.15)	fbs neg. (-0.14)
vader sentiment (0.14)	consonants (-0.13)
verb num. (0.14)	anew arousal (-0.13)
pers. pron. (0.12)	prepositions (-0.12)
total pronouns (0.11)	fricatives (-0.11)
	3rd per. pp sg. (-0.11)
	avg. preverb len. (-0.11)
	nrc disgust (-0.1)

Table 3: The statistical significant features (p<0.1) with at least a small effect size (r>0.1) for the elAFD and enAFD datasets. The features are in ascending p value order. We also report the effect size. Features with moderate effect size (r>0.3) are bold, while common features between the datasets are underlined. pp denotes personal pronouns.

5 Classification

We evaluated the predictive performance of different feature sets and approaches for AFD datasets, including logistic regression experiments[7] and fine-tuned monolingual BERT models for each language[8] (Devlin et al., 2019; Koutsikakis et al., 2020). We also performed cross lingual experiments by exploiting the multilingual BERT model (mBERT) to examine if there are similarities among AFD datasets captured by the BERT.

A stratified split to the datasets was used to create training, testing, and validation subsets with a 70-20-10 ratio. For the cross lingual experiment we trained and validated a model over the 80% and 20% of a language specific dataset respectively, and then tested the performance of the model over the other dataset. We report the results on test sets, while validation subsets were used for fine-tuning the hyper-parameters of the algorithms. For the logistic regression the tuned through brute force parameters were: a) Weka algorithm ($SimpLog|Log$: *simple logistic* (Landwehr et al., 2005) or *logistic* (Le Cessie and Van Houwelingen, 1992)) b) all n-grams of size in $[a, b]$, with $a \geq b$ and $a, b \in [1, 3]$ ((a, b)), c) stemming ($stem$), d) attribute selection ($attrsel$) (applicable only to Log algorithm since it is the de-

[6]https://aprilfoolsdayontheweb.com/2004.html
[7]We employ the Weka API (Hall et al., 2009)
[8]We used tensorflow 2.2.0, keras 2.3.1, and the bert-for-tf2 0.14.4 implementation of google-research/bert, over an AMD Radeon VII card and the ROCm 3.7 platform.

fault for $SimpLog$), e) stopwords removal ($stop$) and, f) lowercase conversion ($lowercase$). For the BERT experiments, the hyperparameters were tuned by random sampling 60 combinations of values, keeping the combination that gave the minimum validation loss. Early stopping with patience 4 was used and the max epochs number was set to 20. The tuned hyperparameters were: learning rate, batch size, dropout rate, max token length, and randomness seeds.

In all cases, we report Recall (R), Precision (P), F-measure (F), Accuracy (A) and AUC (A'). Since the datasets are balanced the majority baseline is 50%. The input for the models consists of the concatenation of the title, the subtitle, the body of the articles and the caption text. Since titles are important for deception detection (Horne and Adali, 2017) and BERT processes texts of up to 512 wordpieces, we placed the title first.

5.1 Logistic Regression Experiments

The examined features sets were: a) the features presented in section 4 (ling), b) n-grams features i.e., phoneme-gram (ph-gram), character-gram (char-gram), word-gram (w-gram), POS-gram (pos-gram), and syntactic-gram (sn-gram) (the latter for the enAFD only), and c) the linguistic+ model that represents the best model that combines the linguistic features with any of the n-gram features. The results are presented in Tables 4 and 5. With * we mark the setups with a statistically significant difference to the best setup regarding accuracy, based on a two proposition z-test (1-tailed) with a 99% CI. We observe that the combination of *lingustic* features with uni/bi/trigrams for the elAFD dataset and the unigrams for the enAFD are the best setups. For the enAFD dataset, the second best model is the combination of *linguistic* features with trigrams. $SimpLog$ seems to perform better, while stemming, lowercase conversion and stopwords removal are generally beneficiary.

5.2 BERT Experiments

In these experiments, we fine-tuned BERT by adding a task-specific linear classification layer on top, using the sigmoid activation function. We also combined BERT with linguistics features by concatenating the embedding of the [CLS] token with the linguistic features, and pass the resulting vector to the task-specific classifier (with a slightly modified architecture). The results of the experiments are presented in Table 6. Although it outperformed logistic regression experiments in both datasets, the differences are not statistical significant. In addition, the combination with linguistic features is not beneficial. Multilingual BERT models perform worse, especially for Greek. In the cross lingual experiments the classifiers performance is limited to about 60% accuracy in both experiments, showcasing that the BERT layers are not able to capture language agnostic information from our datasets.

Best setup	**R**	**P**	**F**	**A'**	**A**
ling.$_{SimpLog}$	62	76	68	82	71
ph-gram$_{(1,2),attrsel,Log}$*	70	67	68	77	68
char-gram$_{(3,3),SimpLog}$*	72	68	70	76	69
w-gram$_{(1,2),SimpLog}$	68	73	71	80	72
pos-gram$_{(2,3),SimpLog}$*	72	65	68	75	67
ling.+$_{word,(1,3),stop,lowercase,SimpLog}$	**74**	**79**	**76**	**85**	**77**

Table 4: Logistic regression results for elAFD.

Best setup	**R**	**P**	**F**	**A'**	**A**
ling.$_{Log}$*	66	80	72	**87**	75
ph-gram$_{(1,1),SimpLog}$	80	77	78	84	78
char-gram$_{(1,3),attrsel,Log}$*	76	72	74	80	73
w-gram$_{(1,1),stem,SimpLog}$	**79**	**81**	**80**	**87**	**80**
pos-gram$_{(3,3),SimpLog}$*	71	69	70	76	69
sn-gram$_{(2,2),SimpLog}$*	80	68	73	77	71
ling.+$_{Word,(3,3),stop,lowercase,SimpLog}$	74	80	77	**87**	78

Table 5: Logistic regression results for enAFD.

	R	**P**	**F**	**A'**	**A**
el$_{bert}$	85	70	77	79	**79**
el$_{bert+ling}$	68	83	75	77	77
el$_{mbert}$	16	57	25	52	52
el$_{mbert+ling}$	62	78	69	72	72
en$_{bert}$	79	86	82	83	**83**
en$_{bert+ling}$	69	87	77	79	79
en$_{mbert}$	37	97	54	68	68
en$_{mbert+ling}$	50	95	66	74	74
en→el $_{mbert}$	31	73	44	60	60
el→en $_{mbert}$	22	84	35	59	59

Table 6: BERT models evaluation results.

6 Conclusion and Future Work

We introduced a new dataset with AFD news articles in Greek and analyzed and compared its deception cues with another English one. The results showcased the use of emotional language, especially of positive sentiment, for deceptive articles which is even more prevalent in the individualistic English dataset. Further, deceptive articles use less concrete language, as manifested by the increased use of adverbs, hedges, and boosters and less usage of named entities, spatial related words and conjunctions compared to the truthful ones. The future and past tenses were correlated with deceptive and truthful articles respectively. All the above, mainly align with previous work (Papantoniou et al., 2021), except from some differences in the usage of pronouns for the Greek dataset, which is attributed to the idiosyncrasies of the news domain. The accuracy of the deployed classifiers offered adequate performance, with no statistically significant differences between the best logistic regression and the BERT models.

In the future we aim at creating even more crosslingual datasets for deception detection tasks through crowdsourcing and by employing the Chattack platform (Smyrnakis et al., 2021).

Acknowledgement

This work has received funding by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the "1st Call for H.F.R.I. Research Projects to support Faculty Members & Researchers and the Procurement of High-and the procurement of high-cost research equipment grant" (Project Number:4195).

References

Edward Dearden and Alistair Baron. 2019. Fool's Errand: Looking at April Fools Hoaxes as Disinformation through the Lens of Deception and Humour. April. 20th International Conference on Computational Linguistics and Intelligent Text Processing, CICLing 2019 ; Conference date: 07-04-2019 Through 13-04-2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–86, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Eileen Fitzpatrick and Joan Bachenko. 2012. Building a Data Collection for Deception Research. In *Proceedings of the Workshop on Computational Approaches to Deception Detection*, pages 31–8, Avignon, France, April. Association for Computational Linguistics.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA data mining software: an update. *SIGKDD Explorations*, 11(1):10–18.

Valerie Hauch, Iris Blandón-Gitlin, Jaume Masip, and Siegfried L. Sporer. 2015. Are Computers Effective Lie Detectors? A Meta-Analysis of Linguistic Cues to Deception. *Personality and Social Psychology Review*, 19(4):307–342. PMID: 25387767.

Geert Hofstede. 1980. *Culture's consequences: International differences in work-related values*. Sage Publications.

Benjamin D. Horne and Sibel Adali. 2017. This Just In: Fake News Packs a Lot in Title, Uses Simpler, Repetitive Content in Text Body, More Similar to Satire than Real News. *ArXiv*, abs/1703.09398.

Marcia K. Johnson, Julie G. Bush, and Karen J. Mitchell. 1998. Interpersonal Reality Monitoring: Judging the Sources of Other People's Memories. *Social Cognition*, 16(2):199–224.

Bennett Kleinberg, Isabelle van der Vegt, Arnoud Arntz, and Bruno Verschuere. 2019. Detecting deceptive communication through linguistic concreteness, Mar.

Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. PHEME dataset for Rumour Detection and Veracity Classification.

Dimitra Koutsantoni. 2005. Greek Cultural Characteristics and Academic Writing. *Journal of Modern Greek Studies*, 23:97–138, 05.

John Koutsikakis, Ilias Chalkidis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2020. GREEK-BERT: The Greeks Visiting Sesame Street. In *11th Hellenic Conference on Artificial Intelligence*, SETN 2020, page 110–117, New York, NY, USA. Association for Computing Machinery.

Niels Landwehr, Mark Hall, and Eibe Frank. 2005. Logistic Model Trees. *Machine Learning*, 59(1):161–205, May.

S. Le Cessie and J.C. Van Houwelingen. 1992. Ridge Estimators in Logistic Regression. *Applied Statistics*, 41(1):191–201.

Jaume Masip, Siegfried L. Sporer, Eugenio Garrido, and Carmen Herrero. 2005. The detection of deception with the reality monitoring approach: a review of the empirical evidence. *Psychology, Crime & Law*, 11(1):99–122.

Galit Nahari, Aldert Vrij, and Ronald P. Fisher. 2014. The Verifiability Approach: Countermeasures Facilitate its Ability to Discriminate Between Truths and Lies. *Applied Cognitive Psychology*, 28(1):122–128.

Matthew L. Newman, James W. Pennebaker, Diane S. Berry, and Jane M. Richards. 2003. Lying Words: Predicting Deception from Linguistic Styles. *Personality and Social Psychology Bulletin*, 29(5):665–75. PMID: 15272998.

Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T. Hancock. 2011. Finding Deceptive Opinion Spam by Any Stretch of the Imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 309–19, Stroudsburg, PA, USA. Association for Computational Linguistics.

Katerina Papantoniou, Panagiotis Papadakos, Theodore Patkos, Giorgos Flouris, Ion Androutsopoulos, and Dimitris Plexousakis. 2021. Deception detection in text and its relation to the cultural dimension of individualism/collectivism. *Natural Language Engineering*. Also appeared as an arXiv preprint arXiv:2105.12530.

Verónica Pérez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. 2018. Automatic Detection of Fake News. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3391–3401, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Denis Peskov, Benny Cheng, Ahmed Elgohary, Joe Barrow, Cristian Danescu-Niculescu-Mizil, and Jordan Boyd-Graber. 2020. It Takes Two to Lie: One to Lie, and One to Listen. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3811–3854, Online, July. Association for Computational Linguistics.

Danae Pla Karidi, Harry Nakos, and Yannis Stavrakas. 2019. Automatic Ground Truth Dataset Creation for Fake News Detection in Social Media. In Hujun Yin, David Camacho, Peter Tino, Antonio J. Tallón-Ballesteros, Ronaldo Menezes, and Richard Allmendinger, editors, *Intelligent Data Engineering and Automated Learning – IDEAL 2019*, pages 424–436, Cham. Springer International Publishing.

Stephen Skalicky, Nicholas Duran, and Scott A Crossley. 2020. Please, Please, Just Tell Me: The Linguistic Features of Humorous Deception. *Dialogue & Discourse*, 11(2):128–149, December.

Emmanouil Smyrnakis, Katerina Papantoniou, Panagiotis Papadakos, and Yannis Tzitzikas. 2021. Chattack: A Gamified Crowd-sourcing Platform for Tagging Deceptive & Abusive Behaviour. In *European Conference on Information Retrieval*, pages 549–553. Springer.

Felix Soldner, Verónica Pérez-Rosas, and Rada Mihalcea. 2019. Box of Lies: Multimodal Deception Detection in Dialogues. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1768–1777, Minneapolis, Minnesota, June. Association for Computational Linguistics.

John Sweller. 2011. Chapter Two - Cognitive Load Theory. volume 55 of *Psychology of Learning and Motivation*, pages 37–76. Academic Press.

Paul J. Taylor, Samuel Larner, Stacey M. Conchie, and Tarek Menacere. 2017. Culture moderates changes in linguistic self-presentation and detail provision when deceiving others. *Royal Society Open Science*, 4(6):170128, June.

Harry C. Triandis and Vasso Vassiliou. 1972. Interpersonal influence and employee selection in two cultures. *Journal of Applied Psychology*, 56:140–145.

Udo Undeutsch, 1989. *The Development of Statement Reality Analysis*, pages 101–19. Springer Netherlands, Dordrecht.

William Yang Wang. 2017. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 422–426. Association for Computational Linguistics.

Visualization: The Missing Factor in Simultaneous Speech Translation

Sara Papi[1,2], **Matteo Negri**[1], **Marco Turchi**[1]
1. Fondazione Bruno Kessler, Italy
2. University of Trento, Italy
{spapi,negri,turchi}@fbk.eu

Abstract

Simultaneous speech translation (SimulST) is the task in which output generation has to be performed on partial, incremental speech input. In recent years, SimulST has become popular due to the spread of multilingual application scenarios, like international live conferences and streaming lectures, in which on-the-fly speech translation can facilitate users' access to audio-visual content. In this paper, we analyze the characteristics of the SimulST systems developed so far, discussing their strengths and weaknesses. We then concentrate on the evaluation framework required to properly assess systems' effectiveness. To this end, we raise the need for a broader performance analysis, also including the user experience standpoint. We argue that SimulST systems, indeed, should be evaluated not only in terms of quality/latency measures, but also via task-oriented metrics accounting, for instance, for the visualization strategy adopted. In light of this, we highlight which are the goals achieved by the community and what is still missing.

1 Introduction

Simultaneous speech translation (SimulST) is the task in which the translation of a source language speech has to be performed on partial, incremental input. This is a key feature to achieve low latency in scenarios like streaming conferences and lectures, where the text has to be displayed following as much as possible the pace of the speech. SimulST is indeed a complex task in which the difficulties of performing speech recognition from partial inputs are exacerbated by the problem to project meaning across languages. Despite the increasing demand for such a system, the problem is still far from being solved.

So far, research efforts mainly focused on the quality/latency trade-off, i.e. producing high quality outputs in the shortest possible time, balancing the need for a good translation with the necessity of a rapid text generation. Previous studies, however, disregard how the translation is displayed and, consequently, how it is actually perceived by the end users. After a concise survey of the state of the art in the field, in this paper we posit that, from the users' experience standpoint, output visualization is at least as important as having a good translation in a short time. This raises the need for a broader, task-oriented and human-centered analysis of SimulST systems' performance, also accounting for this third crucial factor.

2 Background

As in the case of offline speech translation, the adoption of cascade architectures (Stentiford and Steer, 1988; Waibel et al., 1991) was the first attempt made by the SimulST community to tackle the problem of generating text from partial, incremental input. Cascade systems (Fügen, 2009; Fujita et al., 2013; Niehues et al., 2018; Xiong et al., 2019; Arivazhagan et al., 2020b) involve a pipeline of two components. First, a streaming automatic speech recognition (ASR) module transcribes the input speech into the corresponding text (Wang et al., 2020; Moritz et al., 2020). Then, a simultaneous text-to-text translation module translates the partial transcription into target-language text (Gu et al., 2017; Dalvi et al., 2018; Ma et al., 2019; Arivazhagan et al., 2019). This approach suffers from *error propagation*, a well-known problem even in the offline scenario, where the transcription errors made by the ASR module are propagated to the MT module, which cannot recover from them as it does not have direct access to the audio. Another strong limitation of cascaded systems is the *extra latency* added by the two-step pipeline, since the MT module has to wait until the streaming ASR output is produced.

To overcome these issues, the direct models initially proposed in B[Pleaseinsertintopreamble]rard et al. (2016; Weiss et al. (2017) represent a valid alternative that is gaining increasing traction (Bentivogli et al., 2021). Direct ST models are composed of an encoder, usually bidirectional, and a decoder. The encoder starts from the audio features extracted from the input signal and computes a hidden representation; the decoder transforms this representation into target language text. Direct modeling becomes crucial in the simultaneous scenario, as it reduces the overall system's latency due to the absence of intermediate symbolic representation steps. Despite the data scarcity issue caused by the limited availability of speech-to-translation corpora, the adoption of direct architectures showed to be promising (Weiss et al., 2017; Ren et al., 2020; Zeng et al., 2021), driving recent efforts towards the development of increasingly powerful and efficient models.

3 Architectural Challenges

This section surveys the direct SimulST models developed so far, highlighting strengths and weaknesses of the current architectures and decision policies – i.e. the strategies used by the system to decide whether to output a partial translation or to wait for more audio information. We discuss ongoing research on architectural improvements of encoder-decoder models, as well as popular approaches like offline training and re-translation. All these works concentrate on reducing systems latency, targeting a better quality/latency trade-off.

Encoding Strategy. Few studies (Elbayad et al., 2020a; Nguyen et al., 2021b) tried to improve the encoder part of simultaneous systems. Elbayad et al. (2020a) and Nguyen et al. (2021b) introduced the use of unidirectional encoders instead of standard bidirectional encoders (i.e. the encoder states are not updated after each read action) to speed up the decoding phase. Nguyen et al. (2021b) also proposed an encoding strategy called *Overlap-and-Compensate*, where the encoder exploits extra frames provided from the past that were discarded during the previous encoding step. The segmentation problem is a crucial aspect in SimulST, where the system needs to split a long audio input into smaller chunks (speech frames) in order to process them. Different segmentation techniques can be adopted to extract this information, starting from the easiest one based on fixed time windows (Ma et al., 2020b) to the dynamic ones based on automatically detected word boundaries (Zeng et al., 2021; Chen et al., 2021). Ma et al. (2020b) also studied the dynamic segmentation based on oracle boundaries but they discovered that, in their scenario, it had worse performance compared to that of the fixed segmentation.

Decoding Strategy. Some efforts have been made to improve the decoding strategy as it strongly correlates to the decision policy of simultaneous systems. Speculative beam search, or SBS, (Zheng et al., 2019c) represents the first successful attempt to use beam search in SimulST. This technique consists in hallucinating several prediction steps in the future in order to make more accurate decisions based on the best "speculative" prediction obtained. Also Zeng et al. (2021) integrate the beam search in the decoding strategy, developing the wait-k-stride-N strategy. In particular, the authors bypass output speculation by directly applying beam search, after waiting for k words, on a word stride of size N (i.e., on N words at a time) instead of one single word as prescribed by the standard wait-k. Nguyen et al. (2021a) analyzed several decoding strategies relying on different output token granularities, such as characters and Byte Pair Encoding (BPE), showing that the latter yields lower latency.

Offline or Online training? An alternative approach to simultaneous training is the offline (or full-sentence) training of the system and its subsequent use as a simultaneous one. Nguyen et al. (2021a) explored this solution with an LSTM-based direct ST system, analyzing the effectiveness of different decoding strategies. Interestingly, the offline approach does not only preserve overall performance despite the switch of modality, it also improves system's ability to generate well-formed sentences. These results are confirmed by Chen et al. (2021), who successfully exploit a direct ST system jointly trained in an offline fashion with an

ASR one.

Another point of view: re-translation. Re-translation (Niehues et al., 2016; Niehues et al., 2018; Arivazhagan et al., 2020a; Arivazhagan et al., 2020b) consists in re-generating the output from scratch (e.g. after a fixed amount of time) for as long as new information is received. This approach ensures high quality (the final output is produced with all the available context) and low latency (partial translations can be generated with fixed, controllable delay). This, however, comes at the cost of strong output instability (the so-called *flickering*, due to continuous updates of the displayed translations) which is not optimal from the user experience standpoint. To this end, some metrics have been developed to measure the instability phenomenon, such as the *Erasure* (Arivazhagan et al., 2020b), which measures the number of tokens that were deleted from the emitted translation to produce the next translation.

Decision Policy. In simultaneous settings, the model has to decide, at each time step, if the available information is enough to produce a partial translation – i.e. to perform a *write* action using the information received until that step (audio chunk/s in case of SimulST or token/s in case of simultaneous MT) – or if it has to wait and perform a *read* action to receive new information from the input. Possible decision policies result in different ways to balance the quality/latency trade-off. On one side, more read actions provide the system with larger context useful to generate translations of higher quality. On the other side, this counterbalances the increased, sometimes unacceptable latency. To address this problem, two types of policy have been proposed so far: fixed and adaptive. While *fixed* decision policies look at the number of ingested tokens (or speech chunks, in the speech scenario), in the *adaptive* ones the decision is taken by also looking at the contextual information extracted from the input.

While little research focused on adaptive policies (Gu et al., 2017; Zheng et al., 2019a; Zheng et al., 2020) due to the hard and time-consuming training (Zheng et al., 2019b; Arivazhagan et al., 2019), the adoption of very easy-to-train fixed policies is the typical choice. Indeed, the most widely used policy is a fixed one, called *wait-k* (Ma et al., 2019). Simple yet effective, it is based on waiting for k source words before starting to generate the target sentence, as shown in Table 1.

source	It	was	a	way	that	parents	...
wait-3	-	-	-	Es	ging	um	eine
wait-5	-	-	-	-	-	Es	ging

Table 1: wait-k policy example with $k = \{3, 5\}$

As the original wait-k implementation is based on textual source data, Ma et al. (2020b) adapted it to the audio domain by waiting for k fixed time frames (audio chunks or speech frames) rather than k words. However, this simplistic approach does not consider various aspects of human speech, such as different speech rates, duration, pauses, and silences. In (Ren et al., 2020), the adaptation was done differently, by including a Connectionist Temporal Classification (CTC)-based (Graves et al., 2006) segmentation module that is able to determine word boundaries. In this case, the wait-k strategy is applied by waiting for k pauses between words that are automatically detected by the segmenter. Similarly, Zeng et al. (2021) employed the CTC-based segmentation method but applying a *wait-k-stride-N* policy to allow re-ranking during the decoding phase. The *wait-k-stride-N* model emits more than one word at a time, slightly increasing the latency, since the output is prompted after the stride is processed. This small increase in latency, however, allows the model to perform beam search on the stride, which has been shown to be effective in improving translation quality (Sutskever et al., 2014). Decoding more than one word at a time is the approach also employed by Nguyen et al. (2021a), who showed that emitting two words increases the quality of the translation without any relevant impact on latency. Another way of applying the wait-k strategy was proposed by Chen et al. (2021), where a streaming ASR system is used to guide the direct ST decoding. They look at the ASR beam to decide how many tokens have been emitted within the partial audio segment, hence having the information to apply the original wait-k policy in a straightforward way. An interesting solution is also the one by Elbayad et al. (2020a), who jointly train a direct model across multiple wait-k paths. Once the sentence has been encoded, they optimize the system by uniformly sampling the k value for the decoding step. Even though they reach good performance by using a single-path training with k=7 and a different k value for testing, the multi-path approach proved to be effective. One of its advantages is that no k value has to be specified for the training, which allows to avoid the training from scratch of several models for different values of k.

Retrospective. All the aspects analyzed in this section highlight several research directions already taken by the simultaneous community, which have to be studied more in depth. Among all, the audio or text segmentation strategy clearly emerges as a fundamental factor of simultaneous systems, and the ambivalent results obtained in several studies point out that this aspect has to be better clarified. Moreover, the presence of extensive literature on the wait-k policy shows that it represents one of the topics of greatest interest to the community, which continues to work on it to further improve its effectiveness as it directly impacts on the systems' performance, especially latency. Unfortunately, all these studies focus on the architecture enhancements and decision policies despite the absence of a unique and clear evaluation framework to perform a correct and complete analysis of the system.

4 Evaluation Challenges

A good simultaneous model should produce a high quality translation with reasonable timing, as waiting too long will negatively affect the user experience. Offline MT and ST communities commonly use the well-established BLEU metric (Papineni et al., 2002; Post, 2018) to measure the quality of the output translation, but a simultaneous system also needs a metric that accounts for the time spent by the system to output the partial translation. Simultaneous MT (SimulMT) is the task in which a real-time translation is produced having a partial source text at disposal. Since SimulMT was the first yet easiest simultaneous scenario studied by the community, a set of metrics was previously introduced for the textual input-output translation part.

Latency Metrics for SimulMT. The first metric, the *Average Proportion* (AP), was proposed by Cho and Esipova (2016) and measures the average proportion of source input read when generating a target prediction, that is the sum of the tokens read when generating the partial target. However, AP is not length-invariant, i.e. the value of the metric depends on the input and output lengths and is not evenly distributed on the [0, 1] interval (Ma et al., 2019), making this metric strongly unreliable.

To overcome all these problems, Ma et al. (2019) introduced the *Average Lagging* (AL) that directly describes the lagging behind the ideal policy, i.e. a policy that produces the output exactly at the same time as the speech source. As a downside, Average Lagging is not differentiable, which is, instead, a useful property, especially if the metric is likely to be added in the system's loss computation. For this reason, Cherry and Foster (2019) proposed the *Differential Average Lagging* (DAL), introducing a minimum delay after each operation.

Another way of measuring the lagging is to compute the alignment difficulty of a source-target pair. Hence, Elbayad et al. (2020b) proposed the *Lagging Difficulty* (LD) metric that exploits the use of the `fast-align` (Dyer et al., 2013) tool to estimate the source and target alignments. Then, they infer the reference decoding path and compute the AL metric. The authors claimed the LD to be a realistic measure of the simultaneous translation as it also evaluates how a translation is easy to align considering the context available when decoding.

Latency Metrics for SimulST. The most popular AP, AL and DAL metrics were successively adapted by the SimulST community to the speech scenario by converting, for instance, the number of words to the sum of the speech segment durations, as per (Ma et al., 2020a). Later, Ma et al. (2020b) raised the issue of using computational unaware metrics and proposed computational aware metrics accounting for the time spent by the model to generate the output. Unfortunately, computing such metrics is not easy at all in absence of a unique and reproducible environment that can be used to evaluate the model's performance. To this end, Ma et al. (2020a) proposed *SimulEval*a tool which computes the metrics by simulating a real-time scenario with a server-client scheme. This toolkit automatically evaluates simultaneous translations (both text and speech) given a customizable agent that can be defined by the user and that will depend on the adopted policy. Despite the progress in the metrics for evaluating quality and latency, no studies have been conducted on the effective correlation with user experience. This represents a missing key point in the current evaluation framework landscape, giving rise to the need for a tool that combines quality and latency metrics with application-oriented metrics (e.g., reading speed), which are strongly correlated to the visualization and, as an ultimate goal, to the user experience.

5 The Missing Factor: Visualization

In the previous section, we introduced the most popular metrics used to evaluate the simultaneous systems' performance. These metrics account for the quality and the latency of the system without capturing the user needs. Although many researchers acknowledge the importance of human evaluation, this current partial view can push the community in the wrong direction, in which all the efforts are focused on the quality/latency factors while the problem experienced by the user is of another kind. Indeed, the third factor that matters and strongly influences the human understanding of a – even very good – translation is the *visualization strategy* adopted. The visualization problem and the need to present the text in a readable fashion for the user was only faced in our previous work (Karakanta et al., 2021). In the paper, we raised the need for a clearer and less distracting visualization of the SimulST system's generated texts by presenting them as subtitles (text segmented in lines preserving coherent information). We proposed different visualization strategies to better assess the online display problem, attempting to simulate a setting where human understanding is at the core of our analysis.

Visualization modalities. The standard *word-for-word* visualization method (Ma et al., 2019), in which the words appear sequentially on the screen as they are generated, could be strongly sub-optimal for the human understanding (Romero-Fresco, 2011). Infact, the word-for-word approach has two main problems: *i)* the emission rate of words (some go too fast, some too slow) is irregular and the users waste more time reading the text because their eyes have to make more movements, and *ii)* emission of pieces of text that do not correspond to linguistic units/chunks, requiring more cognitive effort. Moreover, when the maximum length of the subtitle (that depends on the dimensions of the screen) is reached, the subtitle disappears without giving the user enough time to read the last words emitted. As this will negatively impact the user experience, we propose in (Karakanta et al., 2021) to adopt different visualization modes that better accommodate the human reading requirements. We first introduced the *block* visualization mode, for which an entire subtitle is displayed at once (usually one or two lines maximum) as soon as the system has finished generating it. This display mode is the easiest to read for the user because it prevents re-reading phenomena (Rajendran et al., 2013) and unnecessary/excessive eye fixations (Romero-Fresco, 2010), reducing the human effort. However, we discovered that the latency introduced by waiting for an entire subtitle is too high to let this visualization mode be used in many simultaneous scenarios. As a consequence, we proposed the *scrolling lines* visualization mode that displays the subtitles line by line. Every time a new line becomes available, it appears at the bottom of the screen, while the previous (older) line is scrolled to the upper line. In this way, there are always two lines displayed on the screen. To evaluate the performance of the system in the different visualization modes, we also proposed an ad-hoc calculation of the *reading speed* (characters per second or CPS) that correlates with the human judgment of the subtitles (Perego et al., 2010). The reading speed shows how fast a user needs to read in order not to miss any part of the subtitle. The lower the reading speed, the better is the model's output since a fast reading speed increases the cognitive load and leaves less time to look at the image. The scrolling line method offers the best balance between latency and a comfortable reading speed resulting to be the best choice for the simultaneous scenario. On the other hand, this approach requires segmented text (i.e. a text that is divided into subtitles), thus the system needs to be able to simultaneously generate transcripts or translations together with proper subtitle delimiters. However, building a simultaneous subtitling system combines the difficulties of the simultaneous setting with the constraint of having a text formatted in proper subtitles. Since both these research directions are still evolving, a lot of work is required to achieve good results.

The lack of studies on this aspects highlights the shortcomings of the actual SimulST systems, individuating possible improvements that will allow the systems to evolve in a more organic and complete way according to the user needs. Moreover, to completely assess the subtitling scenario, a system has to be able to jointly produce timestamps metadata linked to the word emitted, a task that has not been addressed so far. The need for this kind of system represents an interesting direction to follow for the simultaneous community. In the light of this, the researcher should also take into account the three quality-latency-visualization factors in their analyses. We are convinced that these are the most promising aspects to work on to build the best SimulST system for the audience and that human evaluation has to have a crucial role in future studies. We also believe that interdisciplinary dialogue with other fields such as cognitive studies, media accessibility and human-computer interaction would be very insightful to evaluate SimulST outputs from communicative perspectives (Fantinuoli and Prandi, 2021).

6 Conclusions and Future Directions

SimulST systems have become increasingly popular in recent years and many efforts have been made to build robust and efficient models. Despite the difficulties introduced by the online framework, these models have rapidly improved, achieving comparable results to the offline systems. However, many research directions have not been explored enough (e.g., the adoption of dynamic or fixed segmentation, the offline or the online training). First among all, the visualization strategy that is adopted to display the output of the simultaneous systems is an important and largely under-analyzed aspect of the simultaneous experience. We posit that the presence of application-oriented metrics (e.g., reading speed), which are strongly related to the visualization and, as an ultimate goal, to the user experience, is the factor that misses in the actual evaluation environment. Indeed, this paper points out that BLEU and Average Lagging are not the only metrics that matter to effectively evaluate a SimulST model, even if they are fundamental to judge a correct and real-timed translation. We hope that this will inspire the community to work on this critical aspect in the future.

Acknowledgement

This work has been carried out as part of the project Smarter Interpreting (`https://kunveno.digital/`) financed by CDTI Neotec funds.

References

Naveen Arivazhagan, Colin Cherry, Wolfgang Macherey, Chung-Cheng Chiu, Semih Yavuz, Ruoming Pang, Wei Li, and Colin Raffel. 2019. Monotonic infinite lookback attention for simultaneous machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1313–1323, Florence, Italy, July. Association for Computational Linguistics.

Naveen Arivazhagan, Colin Cherry, Wolfgang Macherey, and George Foster. 2020a. Re-translation versus streaming for simultaneous translation. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 220–227, Online, July. Association for Computational Linguistics.

Naveen Arivazhagan, Colin Cherry, Isabelle Te, Wolfgang Macherey, Pallavi Baljekar, and George Foster. 2020b. Re-translation strategies for long form, simultaneous, spoken language translation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7919–7923. IEEE.

Luisa Bentivogli, Mauro Cettolo, Marco Gaido, Alina Karakanta, Alberto Martinelli, Matteo Negri, and Marco Turchi. 2021. Cascade versus direct speech translation: Do the differences still make a difference? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2873–2887, Online, August. Association for Computational Linguistics.

Alexandre Bérard, Olivier Pietquin, Christophe Servan, and Laurent Besacier. 2016. Listen and Translate: A Proof of Concept for End-to-End Speech-to-Text Translation. In *NIPS Workshop on end-to-end learning for speech and audio processing*, Barcelona, Spain, December.

Junkun Chen, Mingbo Ma, Renjie Zheng, and Liang Huang. 2021. Direct simultaneous speech-to-text translation assisted by synchronized streaming ASR. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4618–4624, Online, August. Association for Computational Linguistics.

Colin Cherry and George Foster. 2019. Thinking slow about latency evaluation for simultaneous machine translation.

Kyunghyun Cho and Masha Esipova. 2016. Can neural machine translation do simultaneous translation?

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, and Stephan Vogel. 2018. Incremental decoding and training methods for simultaneous translation in neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 493–499, New Orleans, Louisiana, June. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia, June. Association for Computational Linguistics.

Maha Elbayad, Laurent Besacier, and Jakob Verbeek. 2020a. Efficient Wait-k Models for Simultaneous Machine Translation. In *Proc. Interspeech 2020*, pages 1461–1465.

Maha Elbayad, Michael Ustaszewski, Emmanuelle Esperança-Rodier, Francis Brunet-Manquat, Jakob Verbeek, and Laurent Besacier. 2020b. Online versus offline NMT quality: An in-depth analysis on English-German and German-English. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5047–5058, Barcelona, Spain (Online), December. International Committee on Computational Linguistics.

Claudio Fantinuoli and Bianca Prandi. 2021. Towards the evaluation of automatic simultaneous speech translation from a communicative perspective. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 245–254, Bangkok, Thailand (online), August. Association for Computational Linguistics.

C. Fügen. 2009. A system for simultaneous translation of lectures and speeches.

Tomoki Fujita, Graham Neubig, S. Sakti, T. Toda, and Satoshi Nakamura. 2013. Simple, lexicalized choice of translation timing for simultaneous speech translation. In *INTERSPEECH*.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 369–376, New York, NY, USA. Association for Computing Machinery.

Jiatao Gu, Graham Neubig, Kyunghyun Cho, and Victor O.K. Li. 2017. Learning to translate in real-time with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1053–1062, Valencia, Spain, April. Association for Computational Linguistics.

Alina Karakanta, Sara Papi, Matteo Negri, and Marco Turchi. 2021. Simultaneous speech translation for live subtitling: from delay to display. In *Proceedings of the 1st Workshop on Automatic Spoken Language Translation in Real-World Settings (ASLTRW)*, pages 35–48, Virtual, August. Association for Machine Translation in the Americas.

Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, Hua Wu, and Haifeng Wang. 2019. STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3025–3036, Florence, Italy, July. Association for Computational Linguistics.

Xutai Ma, Mohammad Javad Dousti, Changhan Wang, Jiatao Gu, and Juan Pino. 2020a. SIMULEVAL: An evaluation toolkit for simultaneous translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 144–150, Online, October. Association for Computational Linguistics.

Xutai Ma, Juan Pino, and Philipp Koehn. 2020b. SimulMT to SimulST: Adapting simultaneous text translation to end-to-end simultaneous speech translation. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 582–587, Suzhou, China, December. Association for Computational Linguistics.

Niko Moritz, Takaaki Hori, and Jonathan Le. 2020. Streaming automatic speech recognition with the transformer model. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6074–6078. IEEE.

Ha Nguyen, Yannick Estève, and Laurent Besacier. 2021a. An empirical study of end-to-end simultaneous speech translation decoding strategies. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7528–7532. IEEE.

Ha Nguyen, Yannick Estève, and Laurent Besacier. 2021b. Impact of Encoding and Segmentation Strategies on End-to-End Simultaneous Speech Translation. In *Proc. Interspeech 2021*, pages 2371–2375.

J. Niehues, T. Nguyen, Eunah Cho, Thanh-Le Ha, Kevin Kilgour, Markus Müller, Matthias Sperber, S. Stüker, and A. Waibel. 2016. Dynamic transcription for low-latency speech translation. In *INTERSPEECH*.

J. Niehues, Ngoc-Quan Pham, Thanh-Le Ha, Matthias Sperber, and A. Waibel. 2018. Low-latency neural speech translation. In *INTERSPEECH*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

Elisa Perego, F. Del Missier, M. Porta, and M. Mosconi. 2010. The cognitive effectiveness of subtitle processing. *Media Psychology*, 13:243––272.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium, October. Association for Computational Linguistics.

Dhevi J. Rajendran, Andrew T. Duchowski, Pilar Orero, Juan Martínez, and Pablo Romero-Fresco. 2013. Effects of text chunking on subtitling: A quantitative and qualitative examination. *Perspectives*, 21(1):5–21.

Yi Ren, Jinglin Liu, Xu Tan, Chen Zhang, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2020. SimulSpeech: End-to-end simultaneous speech to text translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3787–3796, Online, July. Association for Computational Linguistics.

Pablo Romero-Fresco, 2010. *Standing on quicksand: hearing viewers' comprehension and reading patterns of respoken subtitles for the news*, pages 175 – 194. Brill, Leiden, The Netherlands.

Pablo Romero-Fresco. 2011. *Subtitling through speech recognition: Respeaking*. Manchester: St. Jerome.

Frederick W. M. Stentiford and Martin G. Steer. 1988. Machine Translation of Speech. *British Telecom Technology Journal*, 6(2):116–122.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Alex Waibel, Ajay N. Jain, Arthur E. McNair, Hiroaki Saito, Alexander G. Hauptmann, and Joe Tebelskis. 1991. JANUS: A Speech-to-Speech Translation System Using Connectionist and Symbolic Processing Strategies. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing, ICASSP 1991*, pages 793–796, Toronto, Canada, May 14-17.

Chengyi Wang, Yu Wu, Shujie Liu, Jinyu Li, Liang Lu, Guoli Ye, and Ming Zhou. 2020. Low latency end-to-end streaming speech recognition with a scout network. *arXiv preprint arXiv:2003.10369*.

Ron J. Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. Sequence-to-Sequence Models Can Directly Translate Foreign Speech. In *Proceedings of Interspeech 2017*, pages 2625–2629, Stockholm, Sweden, August.

Hao Xiong, Ruiqing Zhang, Chuanqiang Zhang, Zhongjun He, Hua Wu, and Haifeng Wang. 2019. Dutongchuan: Context-aware translation model for simultaneous interpreting. *arXiv preprint arXiv:1907.12984*.

Xingshan Zeng, Liangyou Li, and Qun Liu. 2021. RealTranS: End-to-end simultaneous speech translation with convolutional weighted-shrinking transformer. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2461–2474, Online, August. Association for Computational Linguistics.

Baigong Zheng, Renjie Zheng, Mingbo Ma, and Liang Huang. 2019a. Simpler and faster learning of adaptive policies for simultaneous translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1349–1354, Hong Kong, China, November. Association for Computational Linguistics.

Baigong Zheng, Renjie Zheng, Mingbo Ma, and Liang Huang. 2019b. Simultaneous translation with flexible policy via restricted imitation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5816–5822, Florence, Italy, July. Association for Computational Linguistics.

Renjie Zheng, Mingbo Ma, Baigong Zheng, and Liang Huang. 2019c. Speculative beam search for simultaneous translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1395–1402, Hong Kong, China, November. Association for Computational Linguistics.

Baigong Zheng, Kaibo Liu, Renjie Zheng, M. Ma, Hairong Liu, and L. Huang. 2020. Simultaneous translation policies: From fixed to adaptive. *ArXiv*, abs/2004.13169.

Audience Engagement Prediction in Guided Tours through Multimodal Features

Andrea Amelio Ravelli, Andrea Cimino, Felice Dell'Orletta
Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC–CNR)
ItaliaNLP Lab - *www.italianlp.it*
{andreaamelio.ravelli,andrea.cimino,
felice.dellorletta}@ilc.cnr.it

Abstract

This paper explores the possibility to predict audience engagement, measured in terms of visible attention, in the context of guided tours. We built a dataset composed of Italian sentences derived from the speech of an expert guide leading visitors in cultural sites, enriched with multimodal features, and labelled on the basis of the perceivable engagement of the audience. We run experiments in various classification scenarios and observed the impact of modality-specific features on the classifiers.

1 Introduction

During face-to-face interactions, the average speaker is generally very good at estimating the interlocutor's level of involvement, without the need of an explicit verbal feedback. He/she only needs to interpret visually accessible unconscious signals, such as body postures and movements, facial expressions, eye-gazes. The speaker can understand if the addressee is engaged with the discourse, and continuously fine-tune his/her communication strategy in order to keep the communication channel open and the attention high in the audience.[1]

Understanding of non-verbal feedback is not easy to achieve for virtual agents and robots, but this ability is strategic for enabling more natural interfaces capable of adapting to users. Indeed, perceiving signals of loss of attention (and thus, of engagement) is of paramount importance to design naturally behaving virtual agents, enabled to adjust the communication strategy to keep high the interest of their addressees. That information is also a general sign of the quality of the interaction and, more broadly, of the communication experience. At the same time, the ability to generate engaging behaviors in an agent can be beneficial in terms of social awareness (Oertel et al., 2020).

The objective of developing a natural behaving agent, able to guide visitors along a tour in cultural sites, was at the core of the CHROME Project[2] (Cutugno et al., 2018; Origlia et al., 2018), and the present work is intended in the same direction. More specifically, this paper explores the possibility to predict audience engagement in the context of guided tours, by considering acoustic and linguistic features of the speech of an expert guide leading visitors inside museums.

The paper is organised as follows: Section 2 draws a brief overview of related works in the field of engagement annotation and prediction; Section 3 describes in details the construction of the dataset; Section 4 reports the methodology adopted to extract features specific for both linguistic and acoustic modalities; Section 5 illustrates the set of experiments conducted on the collected data, in terms of classification scenarios and features used; Section 6 gathers final observations and ideas for future works.

Contributions The main contributions in this paper are: i) a novel multimodal Italian dataset with engagement annotation; ii) multiple classification scenarios experiments; iii) impact of modality-specific features on multimodal classification.

[1] Recent studies have shown that the processing of *emotionality* in prosody, facial expressions and speech content is associated in the listeners' brain with enhanced activation of auditory cortices, fusiform gyri and middle temporal gyri, respectively, confirming that emotional states are processed through modality-specific modulation strategies (Regenbogen et al., 2012).

[2] Cultural Heritage Resources Orienting Multimodal Experience. `http://www.chrome.unina.it/`

2 Related Works

With the word engagement we refer to the level of involvement reached during a social interaction, which assumes the shape of a process through the whole communication exchange. More specifically, Poggi (2007) defines the process of social engagement as the value that a participant in an interaction attributes to the goal of being together with the other participant(s) and continuing the interaction. Another definition, adopted by many studies in Human-Robot Interaction (HRI),[3] describes engagement as the process by which interactors start, maintain, and end their perceived connections to each other during an interaction (Sidner et al., 2005).

Observations and annotations of engagement are collected on the basis of visible cues, such as facial expressions and reactions, eye gazes, body movements and postures. The majority of the studies are often conducted on a dyadic base, i.e. focusing on communication contexts involving only two participants, most of the times a human interacting with an agent/robot (Castellano et al., 2009; Sanghvi et al., 2011; Ben-Youssef et al., 2021). Nevertheless, engagement can be measured in groups of people taking part in the same communication event as the average of the degree to which individuals are involved (Gatica-Perez et al., 2005; Oertel et al., 2011). Human-to-human interactions within groups have been studied principally in the research field of education (Fredricks et al., 2004) where visible cues are related to attention, which is considered as a perceivable proxy to the more complex and inner process of engagement (Goldberg et al., 2019).

3 Dataset

The dataset presented in this paper is derived from a subset of the CHROME Project data collection (Origlia et al., 2018), which comprises aligned videos, audios and transcriptions of guided tours in three Charterhouses in Campania. Two videos have been recorded for each session: one video with the guide as subject, the other focused on the group of visitors. Data of 3 visits with the same expert guide (in the same Charterhouse) have been selected. Each visit is organised in 6 points of interest (POI), i.e. rooms or areas inside the Charterhouse where groups stop during the tours and the guide describes the place with its furnishings, history, and anecdotes.

In total, starting data consist of 2:44:25 hours of audiovisual material and 22,621 tokens from the aligned transcriptions. The language of the speech is Italian.

3.1 Annotation and Segmentation

Engagement has been annotated as a continuous measurement of visitor's attention, as a visible cue of engagement. The annotation has been carried out using PAGAN Annotation Tool (Melhart et al., 2019), and performed by two annotators watching videos of the groups of visitors in order to observe cues of gain or loss of attention. Following Oertel et al. (2011), annotators have been asked to evaluate the average behaviour of the whole group. Agreement between the two annotators is consistent, with an average Spearman's rho of 0.87 (Ravelli et al., 2020).

The raw transcriptions have been manually segmented with the objective of creating textual segments close to written sentences, and this segmentation has been projected on audio files, in order to obtain aligned text-audio pairs for each segment. Given that every visit is similarly structured, and also topics and whole pieces of information are mostly the same across different visits, the resulting transcriptions are extremely clear and phenomena such as retracting and disfluencies are minimum if compared to transcriptions of typical spontaneous speech. Thus, text normalisation (i.e., disfluencies removal, basic punctuation insertion) has been easy to obtain, and the resulting adaptation lead to sentences easy to parse with common NLP tools trained on written texts.

Segmentation has been performed on the basis of perceptual cues of utterance completeness. As described by Danieli et al. (2005), a break is said terminal if a competent speaker (i.e. mother tongue speaker) assigns to it the quality of concluding the sequence. Starting with this observation, two annotators have been asked to listen to the original audio tracks and mark transcriptions with a full stop where they perceived a break as a boundary between utterances, on the basis of intonation and prosodic contour. Utterances perceived as independent but pronounced too quickly to allow a clean cut (especially considering audio segmentation and the consequent features extraction) have been kept together in a single segment.

[3] For a broad and complete overview of works on engagement in HRI studies, see Oertel et al. (2020)

To assess the reliability of the segmentation process, we measured the accuracy between the two annotators on a subset of the data (the 40% of the total, corresponding to one of the three visits). We adopted a chunking approach to the problem, by adapting an IOB (Inside-Outside-Begin) tagging framework to label tokens, from the continuous transcriptions of the sample, at the beginning (B), inside (I), end (E) of segments, or outside (O) any of those. We measured an accuracy of 91,53% in terms of agreement/disagreement on the basis of the series of labelled tokens derived for each annotator.

At the end of the segmentation process, the dataset counts 1,114 Italian sentences, with an average of 20.31 tokens per sentence (std: 11.96), and an average duration of audio segments of 8.13 seconds (std: 5.22).

An engagement class has been assigned to each sentence: 1 if an increase in engagement has been recorded in the span of that sentence, 0 in case of decrease or no variation. To compute the class, we considered the delta between the input and output values of the continuous measurement obtained with the annotations, with respect to the beginning and end of sentences. Specifically, for each sentence we selected all the annotations (one per millisecond) falling into the sentence boundaries, and then we subtracted the value of the first one from the last one. We reduced the task to a binary classification in order to test to which extent it is possible to predict engaging content before to evaluate the possibility to expand the analysis to a finer classification, accounting also for what is specifically engaging, not-engaging or neutral.

4 Features Extraction

In order to train and test a classifier in predicting the engagement of the addressee of an utterance, using both linguistic and acoustic information, features specific for each modality have been extracted independently, and then concatenated as unique vectors representing each entry of the dataset.

4.1 Linguistic Features

The textual modality has been encoded by using Profiling–UD (Brunato et al., 2020), a publicly available web–based application[4] inspired to the methodology initially presented in Montemagni (2013), that performs linguistic profiling of a text, or a large collection of texts, for multiple languages. The system, based on an intermediate step of linguistic annotation with UDPipe (Straka et al., 2016), extracts a total of 129 features per each analysed document. In this case, Profiling-UD analysis has been performed per sentence, thus the output has been considered as the linguistic feature set of each segment of the dataset. Table 1 reports the 127 features extracted with Profiling-UD and used as textual modality features for the classifier.[5]

Linguistic features	**n**
Raw text properties	2
Morpho–syntactic information	52
Verbal predicate structure	10
Parsed tree structures	15
Syntactic relations	38
Subordination phenomena	10
Total	**127**

Table 1: Set of linguistic features extracted with Profiling-UD.

4.2 Acoustic Features

The acoustic modality has been encoded using OpenSmile[6] (Eyben et al., 2010), a complete and open-source toolkit for analysis, processing and classification of audio data, especially targeted at speech and music applications such as automatic speech recognition, speaker identification, emotion recognition, or beat tracking and chord detection. The acoustic features set used in this case is the Computational Paralinguistics ChallengE[7] (ComParE), which comprises 65 Low-Level Descriptors (LLDs), computed per frame. Table 2 reports a summary of the ComParE LLDs extracted with OpenSmile, grouped by type: prosody-related, spectrum-related and quality-related.

Given that the duration (and number of frames, consequently) of audio segments varies, common transformations (min, max, mean, median, std) have been applied on the set of per-frame features

[4] Profiling-UD can be accessed at the following link: `http://linguistic-profiling.italianlp.it`

[5] Out of the 129 Profiling-UD features, *n_sentences* and *tokens_per_sent* (raw text properties) have not been considered, given that the analysis has been performed per sentence.

[6] `https://www.audeering.com/research/opensmile/`

[7] `http://www.compare.openaudio.eu`

Acoustic features	n
Prosodic	
F_0 (SHS and viterbi smoothing)	1
Sum of auditory spectrum (loudness)	1
Sum of RASTA-style filtered auditory spectrum	1
RMS energy, zero-crossing rate	2
Spectral	
RASTA-style auditory spectrum, bands 1–26 (0–8 kHz)	26
MFCC 1–14	14
Spectral energy 250–650 Hz, 1 k–4 kHz	2
Spectral roll off point 0.25, 0.50, 0.75, 0.90	4
Spectral flux, centroid, entropy, slope	4
Psychoacoustic sharpness, harmonicity	2
Spectral variance, skewness, kurtosis	3
Sound quality	
Voicing probability	1
Log. HNR, Jitter (local, delta), Shimmer (local)	4
Total	**65**

Table 2: Set of acoustic features extracted with OpenSmile.

of each segment, leading to a total of 325 acoustic features (65 LLDs x 5 transformations).

5 Experiments

To explore the possibility to predict engaging sentences, we implemented a machine learning classifier using the linear SVM algorithm provided by the scikit-learn library (Pedregosa et al., 2011).

We defined various classification scenarios on the basis of 3 different train-test splitting of the dataset. The first, and more common scenario, is based on a *k*-fold setting, in which data has been randomly split in 10 folds, trained on 9 of them and tested on the remaining one. The second scenario uses data from one POI from all the visits as a test, and it is trained on the remaining parts. The third scenario considers data from a whole visit as test and is trained on the remaining two. Global results are obtained by averaging the classification performances of each run per scenario (e.g. average of all *k*-fold outputs tested on every fold).

For each scenario, the SVM classifier has been trained and tested three times, once per single modality (i.e. linguistic or acoustic features exclusively) and once with joint representations (the full set of both linguistic and acoustic features). All the features have been normalised in the range $[0, 1]$ using the *MinMaxScaler* algorithm implemented in scikit-learn.

	***k*-fold**	**POI**	**Visit**
Baseline	51.53%	47.05%	47.32%
Linguistic	**57.81%**	**58.05%**	**57.44%**
Acoustic	55.35%	55.64%	55.83%
Multimodal	53.49%	54.25%	54.40%

Table 3: Accuracy scores for each classification scenario with all features settings.

Table 3 reports the aggregated results, in terms of accuracy, from all the experiments. The baseline considered is the assignment of the majority class found in the training data. All the classifiers in the three scenarios obtain better results than the baseline, but the multimodal systems (the ones exploiting both linguistic and acoustic sets of features) are never able to do better than models based on linguistic features only. Moreover, it is possible to observe that multimodal systems achieve scores similar to acoustic systems.

Low performances, especially for multimodal systems, may be ascribed to the fact that the classifiers are fed with too many features (452 total; 127 textual and 325 acoustic features) with respect to the dimension of the dataset (1,114 items), and thus they build representations with low variation in terms of single feature weight. Moreover, summing the two sets in the multimodal systems leads to worst results than single-modality systems, amplifying the problem.

In order to verify this hypotesis, we reduced the number of features by observing the weights assigned to each feature by classifiers trained on single modalities, and selecting only the top 20 from each ranked set. Figures 1 and 2 show the reduced set of features along with their weights for the linguistic and acoustic set of features, respectively. Among the top-rated, on the linguistic side, we can find features related to the syntactic tree of the sentence and verbal predicate structure; on the acoustic side, principally spectral and prosodic features.

As shown in Table 4, by using this reduced features sets, all systems obtain better results with respect to the experiments conducted exploiting the whole sets of features. Most significant improvements can be traced for models based on acoustic

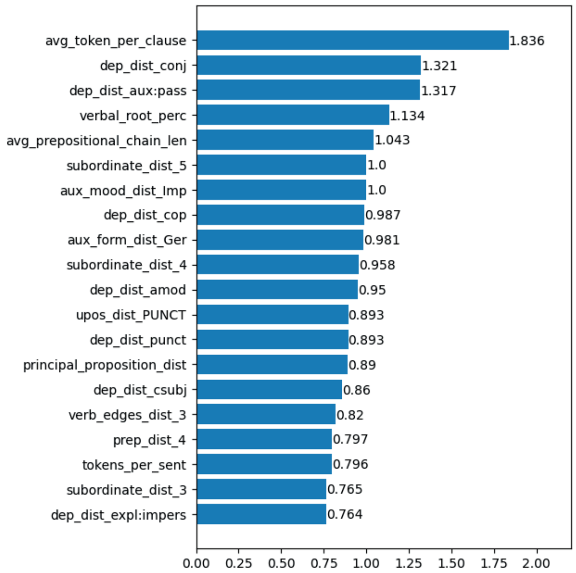


Figure 1: Top 20 linguistic features.

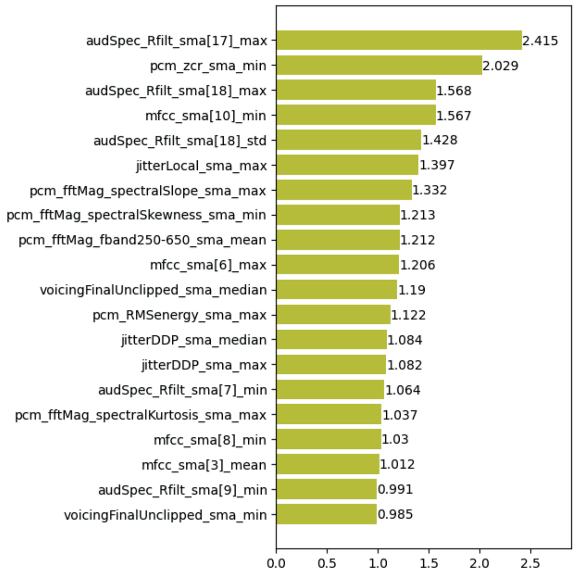


Figure 2: Top 20 acoustic features.

	k-fold	POI	Visit
Baseline	51.53%	47.05%	47.32%
Linguistic	60.78%	59.86%	60.39%
Acoustic	65.70%	63.87%	**64.86%**
Multimodal	**66.07%**	**65.36%**	64.03%

Table 4: Accuracy score for each classification scenario with best features settings.

and multimodal features set, with an average increase in accuracy of the 10%. Differently from previous experiments, multimodal systems reach the best overall results in two out of three scenarios (*k*-fold and POI).

Again, multimodal systems scores are close to those obtained exploiting exclusively acoustic features. For this reason, we compared the predictions from single modalities with multimodal ones, and we found out that multimodal systems predictions overlap more with acoustic systems (0.86) than with linguistic systems (0.79). It confirms that this behaviour is due to the fact that acoustic features are those more considered by the multimodal classifier.

It is possible to observe the higher contribution from acoustic features to the multimodal systems in Figure 3: among the top 10 most important features, only 2 are linguistic, and the trend is dramatically off balance in favour of acoustic features.

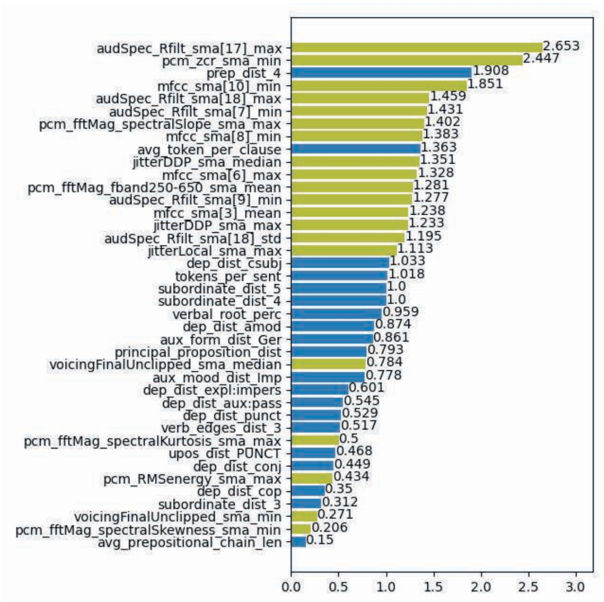


Figure 3: Features weight after selecting best 20 linguistic and acoustic features.

6 Conclusions

In this paper we introduced a novel multimodal dataset for the analysis and prediction of engagement, composed of Italian sentences derived from the speech of an expert guide leading visitors in cultural sites, enriched with multimodal features, and labelled on the basis of the perceivable engagement of the audience. We performed several experiments in different classification scenarios, in order to explore the possibility to predict engagement on the basis of features extracted for both the linguistic and acoustic modalities. Combining modalities in classification leads to good results, but with a filtered set of features to avoid too noisy representations. An in interesting experiment would be to combine the outcomes of two different systems (one exploiting exclusively acoustic features, linguistic features the other) rather than using a monolithic one fed with all the features. This technique often leads to better performances with respect to the decisions taken by a single system (Woźniak et al., 2014; Malmasi and Dras, 2018).

Moreover, we are working on aligning features derived from the visual modality, by encoding information from the videos used to annotate engagement. In this way, the dataset will contain a more complete representation, and it would be possible to correlate perceived engagement in the audience with the full set of stimuli offered during the guided tour.

Acknowledgments

The authors would like to acknowledge the contribution of Luca Poggianti and Mario Gomis, who have annotated the engagement on the videos, and Federico Boggia and Ludovica Binetti, who have segmented the sentences of the dataset.

References

Atef Ben-Youssef, Chloé Clavel, and Slim Essid. 2021. Early detection of user engagement breakdown in spontaneous human-humanoid interaction. *IEEE Transactions on Affective Computing*, 12(3):776–787.

Dominique Brunato, Andrea Cimino, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2020. Profiling-ud: a tool for linguistic profiling of texts. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 7145–7151.

Ginevra Castellano, André Pereira, Iolanda Leite, Ana Paiva, and Peter W McOwan. 2009. Detecting user engagement with a robot companion using task and social interaction-based features. In *Proceedings of the 2009 international conference on Multimodal interfaces*, pages 119–126.

Francesco Cutugno, Felice Dell'Orletta, Isabella Poggi, Renata Savy, and Antonio Sorgente. 2018. The CHROME Manifesto: Integrating Multimodal Data into Cultural Heritage Resources. In Elena Cabrio, Alessandro Mazzei, and Fabio Tamburini, editors, *Proceedings of the Fifth Italian Conference on Computational Linguistics, CLiC-it 2018, Torino, Italy, December 10-12 , 2018*, volume 2253 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Morena Danieli, Juan María Garrido, Massimo Moneglia, Andrea Panizza, Silvia Quazza, and Marc Swerts. 2005. Evaluation of Consensus on the Annotation of Prosodic Breaks in the Romance Corpus of Spontaneous Speech "C-ORAL-ROM". In Emanuela Cresti and Massimo Moneglia, editors, *C-ORAL-ROM: integrated reference corpora for spoken romance languages*, pages 1513–1516.

Florian Eyben, Martin Wöllmer, and Björn Schuller. 2010. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1459–1462.

Jennifer A Fredricks, Phyllis C Blumenfeld, and Alison H Paris. 2004. School engagement: Potential of the concept, state of the evidence. *Review of educational research*, 74(1):59–109.

Daniel Gatica-Perez, L McCowan, Dong Zhang, and Samy Bengio. 2005. Detecting group interest-level in meetings. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 1, pages I–489. IEEE.

Patricia Goldberg, Ömer Sümer, Kathleen Stürmer, Wolfgang Wagner, Richard Göllner, Peter Gerjets, Enkelejda Kasneci, and Ulrich Trautwein. 2019. Attentive or Not? Toward a Machine Learning Approach to Assessing Students' Visible Engagement in Classroom Instruction. *Educational Psychology Review*, 35(1):463–23.

Shervin Malmasi and Mark Dras. 2018. Native language identification with classifier stacking and ensembles. *Comput. Linguistics*, 44(3).

David Melhart, Antonios Liapis, and Georgios N Yannakakis. 2019. PAGAN: Video Affect Annotation Made Easy. In *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 130–136. IEEE.

Simonetta Montemagni. 2013. Tecnologie linguistico-computazionali e monitoraggio della lingua italiana. *Studi Italiani di Linguistica Teorica e Applicata (SILTA)*, XLII(1):145–172.

Catharine Oertel, Stefan Scherer, and Nick Campbell. 2011. On the use of multimodal cues for the prediction of degrees of involvement in spontaneous conversation. In *Twelfth annual conference of the international speech communication association*.

Catharine Oertel, Ginevra Castellano, Mohamed Chetouani, Jauwairia Nasir, Mohammad Obaid, Catherine Pelachaud, and Christopher Peters. 2020. Engagement in human-agent interaction: An overview. *Frontiers in Robotics and AI*, 7:92.

Antonio Origlia, Renata Savy, Isabella Poggi, Francesco Cutugno, Iolanda Alfano, Francesca D'Errico, Laura Vincze, and Violetta Cataldo. 2018. An audiovisual corpus of guided tours in cultural sites: Data collection protocols in the chrome project. In *2018 AVI-CH Workshop on Advanced Visual Interfaces for Cultural Heritage*, volume 2091.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Isabella Poggi. 2007. *Mind, hands, face and body: a goal and belief view of multimodal communication*. Weidler.

Andrea Amelio Ravelli, Antonio Origlia, and Felice Dell'Orletta. 2020. Exploring Attention in a Multimodal Corpus of Guided Tours. In Johanna Monti, Felice Dell'Orletta, and Fabio Tamburini, editors, *Proceedings of the Seventh Italian Conference on Computational Linguistics, CLiC-it 2020, Bologna, Italy, March 1-3, 2021*, volume 2769 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Christina Regenbogen, Daniel A Schneider, Raquel E Gur, Frank Schneider, Ute Habel, and Thilo Kellermann. 2012. Multimodal human communication — Targeting facial expressions, speech content and prosody. *NeuroImage*, 60(4):2346–2356.

Jyotirmay Sanghvi, Ginevra Castellano, Iolanda Leite, André Pereira, Peter W. McOwan, and Ana Paiva. 2011. Automatic analysis of affective postures and body motion to detect engagement with a game companion. In *Proceedings of the 6th International Conference on Human-Robot Interaction*, HRI '11, page 305–312, New York, NY, USA. Association for Computing Machinery.

Candace L Sidner, Christopher Lee, Cory D Kidd, Neal Lesh, and Charles Rich. 2005. Explorations in engagement for humans and robots. *Artificial Intelligence*, 166(1-2):140–164.

Milan Straka, Jan Hajič, and Jana Straková. 2016. UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4290–4297, Portorož, Slovenia. European Language Resources Association (ELRA).

Michał Woźniak, Manuel Graña, and Emilio Corchado. 2014. A survey of multiple classifier systems as hybrid systems. *Information Fusion*, 16:3–17. Special Issue on Information Fusion in Hybrid Intelligent Fusion Systems.

FANCY: A Diagnostic Data-Set for NLI Models

Guido Rocchietti, Flavia Achena, Giuseppe Marziano, Sara Salaris, Alessandro Lenci
University of Pisa, Italy
guido.rocchietti@gmail.com, flavia.achena@gmail.com,
marzianogiuseppe@gmail.com, sarasalaris16@gmail.com,
alessandro.lenci@unipi.it

Abstract

We present here FANCY (FActivity, Negation, Common-sense, hYpernimy), a new dataset with 4000 sentence pairs concerning complex linguistic phenomena such as factivity, negation, common-sense knowledge, hypernymy and hyponymy. The analysis is developed on two levels: coarse-grained for the labels of the Natural Language Inference (NLI), that is to say the task of determining whether a hypothesis is true (entailment), false (contradiction), or undetermined (neutral) and fine-grained for the linguistic features of each phenomenon. For our experiments, we analyzed the quality of the sentence embeddings generated from two transformer-based neural models, BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019b), that were fine-tuned on MNLI and were tested on our dataset, using CBOW as a baseline. The results obtained are lower than the performance of the same models on benchmarks like GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019) and allow us to understand which linguistic features are the most difficult to understand.

1 Introduction

Nowadays it has become more and more important to understand how much neural models applied to Natural Language Processing can understand about language features.

The **probing task** methodology is a simple but effective approach to address this issue (Conneau et al., 2018). A network is trained on a specific task and then the representations are passed to a classifier. The performance of the classifier is evaluated with a dataset constructed to test the understanding of specific linguistic phenomena. If the classifier performs well, then it can be deduced that the neural embeddings have stored syntactic and semantic knowledge relative to those specific linguisitc phenomena.

One of the most widely used tasks for this approach is Natural Language Inference, in which the model must decide whether a *hypothesis* is an entailment, a contradiction, or simply neutral with respect to the *premise*.

Another approach consists in using benchmarks, i.e. datasets relating to various types of tasks, which are able, on the basis of the results obtained, to provide a general judgment on the performance of the model. Although benchmarks are very useful in evaluating the average performance of models, they are less effective in representing a wide range of linguistic phenomena that the models are able to deal with.

It is in this context that the *challenge sets* are born, (also called *adversarial sets*, *stress sets* or *diagnostic sets*) such as the SNLI (Stanford Natural Language Inference) (Bowman et al., 2015) and the MultiNLI (Multi-genre Natural Language Inference) (Williams et al., 2018). This datasets provide the possibility of more specific evaluation frameworks compared to traditional benchmarks (Belinkov and Glass, 2019): as in the case of the probing task, the aim is to evaluate the quality of linguistic information encoded by vector representations.

For our research we built a diagnostic dataset that addresses key aspects of the human knowledge of lexical and compositional meaning, in order to test the deep semantic abilities of the latest computational models.

In this paper, we introduce FANCY, a dataset with 4,000 different hand-annotated sentence pairs with inference relation between them. In Section 3 we will briefly present the linguistic phenomena we decided to analyze. In Section 4 we will present the methods of dataset construction and in Section 5 we will discuss the results of the experiments conducted on FANCY.

2 Related Work

Despite the progress made in recent years in the study of vector representations, it is still difficult to understand exactly what kind of linguistic properties they capture. The main approaches used in this area are probing tasks and diagnostic datasets.

A probing task is a classification problem focused on the simple linguistic properties of sentences (Conneau et al., 2018). This approach has been used on a wide variety of linguistic phenomena. The work of Ettinger (2016), for example, focused on semantic role and negation scope: the sentence embeddings used are Skip-Thought (Kiros et al., 2015), Paragram (Le and Mikolov, 2014) and those obtained from the average of GloVe word embeddings (Pennington et al., 2014). Adi et al. (2016) verified whether sentence embeddings are able to encode information such as the order, length and content of words in a sentence. These elements were evaluated on sentence embedding produced by CBOW (Continuous Bag-of-Words) and Encoder-Decoder (ED) models, both pre-trained on Wikipedia.

On the other hand, the importance of challenge sets is demonstrated by the fact that some traditional benchmarks have been equipped, in addition to the standard datasets, with challenge sets dedicated entirely to the NLI task. In fact, both GLUE and SuperGLUE have a diagnostic dataset, consisting of about 1000 pairs of manually constructed sentences involving 30 linguistic phenomena, including anaphora, factivity, negation, redundancy, hyponymy, etc. Similar challenge sets have been developed and described in the publications of Naik et al. (2018), a dataset in which the errors committed related to negation, antinomies and numerical reasoning are also investigated, Glockner et al. (2018), a challenge set created with particular reference to common knowledge and McCoy et al. (2019), an evaluation dataset that contains 30,000 specific examples on which neural models perform incorrect classifications, such as lexical overlap, subsequence, constituent, etc.

3 Linguistic Phenomena

We selected four different kinds of linguistic phenomena to analyze: (1) the *factivity*, which address the truthfulness and the factuality of the events mentioned inside the phrases, (2) the *negation*, which in the English language can be expressed by several terms and situations, (3) *hierarchical relations*, i.e. semantic relations like hypernymy between a general term and a more specific term, and (4) the *common-sense knowledge*, which relates to the shared knowledge among speakers about events and facts concerning the real world.

3.1 Factivity

Factivity is a linguistic phenomenon related to the truthfulness of events or concepts that are mentioned and expressed in a sentence: each event, based on the elements contained in the sentence, can assume a certain degree of certainty.

a. John **thinks** it's raining.
b. John **knows** it's raining.

When a speaker reads the non-factive verb *think* (a.), he understands that the event mentioned in the sentence (*it's raining*) is just a possibility, while he deduces that it's a fact when the factive verb *know* is used (b.).

When we talk of situations and events that occur, have occurred or will surely occur in the world, we present them as facts, while we usually complete our tales using approximations in cases where we do not know whether the things we are talking about have actually happened and we are not completely sure of their certainty. It is in this context that we can observe the phenomenon of *factivity* (Saurí and Pustejovsky, 2012).

3.2 Negation

Negation is a complex phenomenon that characterize human language among all (Horn, 1989). From a logical perspective, it is the opposite of affirmation, which means that the truth value of the statement is reversed by the negative. The main challenge is to identify the *scope* of the negative marker within the sentence, i.e. which element is semantically negated (Jackendoff, 1969). If we consider a sentence such as *Mary does not read carefully*, we can observe that the scope is partial, because the negation refers only to the adverb. Besides the most common *not*, *nobody* and *nothing*, we have taken into account all possible negative cases in the English language.

Negation may be implicit, such as *forget* meaning *not remember*, or affixal in such terms as *illegal* or *dis-agreement*. It could be related to quantifiers, in cases such as *not all veggies are tasty* which contradicts *all veggies are not tasty*. Some sentences can occur with double negative markers, such as *John called neither his father nor his mother*. Moreover, we can observe contrastive negation (McCawley, 1991), in sentences like *John drank not coffee but tea*. So, although characteristic of all languages and frequently used, negation is a complex phenomenon to investigate.

3.3 Hierarchical Relations

In many cases, the entailment relations can occur not only at a sentence level but also at a word level, if we consider the meaning relations that exist between words: these kinds of relations are defined as lexical entailment (Roller, 2017) and they are determined for example by *subtype/type* hierarchical relations such as *hyponymy* (*dog* is hyponym of *animal*) and *troponymy* (*run* is troponym of *move*) (Pustejovsky and Batiukova, 2019). We define the *subtype/type* relation as entailment (*dog* entails *animal*) and the *type/subtype* relation as neutral (*animal* does not entail *dog*) (MacCartney and Manning, 2009). However, the logical relations between lexical elements can be differently projected by the properties (upward monotone, downward monotone and non-monotone) of some semantic functions (*projectivity signatures*) such as restrictive quantifiers (some, any, every, etc.), negation and superlative (MacCartney and Manning, 2014). A function is *upward monotone* if the logical relation between premise and hypothesis is projected without change: the sentence *some parrots talk* entails *some birds talk*. A function is *downward monotone* if it reverses the logical relations between premise and hypothesis: *no fish talk* entails *no carp talk*. A function is *non-monotone* if it projects the logical relation between premise and hypothesis as neutral: *most humans talk* does not entail *most animals talk* (and vice-versa).

3.4 Common-Sense Knowledge

The concept of common-sense is hard to define because it is strictly entangled with the way we humans reason. Even though its definition is controversial, we adopt here what Feldman called *The Standard View* (Feldman, 2003). In his book he defined eleven categories that give us an idea of the things we know as human beings. He stated two different thesis that constitute the Standard View: the first one states that *We know a large variety of things in categories (a)-(k)*[1] and the second one states that *Our primary sources of knowledge are (a)-(f)*[2].

Starting from the types suggested by LoBue and Yates (2011), we grouped common-sense into five macro-categories.

Causal Relations The categories in which the statement of the premise causes the hypothesis statement, e.g. *the man had a bath* entails *the man got wet*: here we can see how the fact that the man took a bath is the cause for him of being wet, hence there is a *Cause/Effect* relation. At the same time the fact that *Mary was married to John* automatically implies *John was married to Mary*, therefore the relation is of *Simultaneous Condition*.

Spatial Relations This category includes sentences that specify the physical position of an agent or an object with respect to someone or something, e.g. the fact that *John is inside his home* contradicts the sentence *John is close to his home* because: in this case, the spatial prepositions *inside* and *close to* cannot subsist at the same time.

Temporal Relations In this category are included texts that specify the time of en event with respect to someone or something, e.g. the fact that *Julius Caesar was assassinated in 44 B.C.* implies that *Julius Caesar died before the birth of Christ*. In this example the reader is supposed to know that B.C. indicates the birth of Christ, which is not trivial.

World Knowledge Relations All the categories that suppose a previous knowledge of the phenomenal or human world, for example all the sentences that suppose a geographic knowledge to be correctly tagged, e.g. *Charles Dickens is buried in Westminster Abbey* implies that *Charles Dickens rests in London* only if we know that Westminster is in London.

Other Relations In this set we put all the categories which are not included in the previous ones (e.g., arithmetic relations and mutually exclusive relations). For example, *On the train, there are 340 passengers and 40 employees* implies that *On the train, there are 380 people* because we know that if there are 340 + 40 people on the train then the total of the people will be 380.

[1]The categories that we know, such as the past, morality, science etc.

[2]He individuated six different sources of knowledge such as perception, memory,reasoning etc.

4 Dataset Construction

The dataset created for the experiments consists in 4000 pairs of sentences that were built manually by the authors, and this is because we decided to only include sentences that were as simple and clear as possible, in order to specifically focus on the linguistic features of the phenomena and to exclude other external factors of complexity that could have affected the performance of the neural models. For the construction of FANCY, we followed the diagnostic dataset schema provided with the SuperGlue[3] benchmark for models evaluation, so all the data were inserted in a tabular framework and tagged with the following columns and labels.

Premise and Hypothesis Are the first two columns of the dataset and indicate which sentence is the premise and which is the hypothesis.

FW and BW These two columns point out which one of the sentences should be used as the premise. For instance, if we find the sentence *Granada is in Spain* as the premise, and *Granada is in Europe* as the hypothesis in the database, the column FW (forward) considers the first as the *premise* and the second as the *hypothesis* while the columns BW (backward) considers the second sentence as the premise and the first as the hypothesis. In both of the columns we inserted the correct output: in the example above, the column FW would contain the tag *entailment*, because the first sentence implies the second one, while the column BW would contain the tag *neutral* because the second sentence does not imply the first one but does not contradict that either.

Phenomenon Category This column is very important for this study because it specifies which kind of feature regarding a particular phenomenon is represented by the sentence pairs.

Phenomenon	E	N	C
Factivity	239	465	296
Negation	410	428	158
Hierarchical	369	475	156
Common-sense	388	254	358

Table 1: Distribution of Entailment (E), Neutral (N) and Contradiction (C) labels.

In Table 1 we can see that FANCY is composed of **1406** pairs of sentences that lead to an entailment, **1622** sets of neutral sentences and **968** contradictions.

[3]https://super.gluebenchmark.com/diagnostics

5 Experiments

In this section, we report the results of the experiments conducted using our dataset FANCY. We tested state-of-the-art models for NLI on the four different linguistic phenomena in the dataset. We selected *bert-base-uncased-MNLI* and *roberta-large-mnli*, both of which were finetuned on the MNLI dataset, and also a baseline model based on CBOW. The BERT and RoBERTa models are based on the Transformer architecture and are available on the Hugging Face web page.[4] For what concerns the CBOW model, it was built using the tensorflow library,[5] with the word embeddings generated by GloVe pretrained with 840 Billions tokens, a vocabulary of 2.2 millions cased words and the resulting word vectors with 300 dimensions.[6] The model was then trained on the MultiNLI dataset, so that all three models were trained on the same data.

Set	BERT	RoBERTa	CBOW
MNLI	84.6	90.2	65.2
Factivity	65.2	74.6	45.1
Negation	70.0	82.0	45.0
Hierarchical	49.7	60.4	37.8
Common-sense	57.0	68.0	41.0

Table 2: Accuracies report.

We tested every model on the examples of FANCY. The results in Table 5 show how the models struggled to address these kind of phenomena, if compared with the results on the MNLI. We can see that the baseline model performed quite poorly on all the subsets of our data. RoBERTa is the best performing one, even though it showed poor performances on linguistic phenomena such as common-sense and hierarchical relations while performing better on factivity and negations.

Label	Error	Tot	%
Possibly Fact	257	416	62
Possibly Counterfact	8	50	16
Fact	27	244	11
Counterfact	32	290	11

Table 3: RoBERTa errors on factivity relations.

In Table 3 we can see the errors that RoBERTa made in labeling examples regarding *factivity*. Most of the errors concern examples where the *hypothesis* gave place to a *Possible fact* and therefore should be tagged as *neutral*.

[4]https://huggingface.co/

[5]https://www.tensorflow.org/

[6]https://nlp.stanford.edu/projects/glove/

Premise	Hypothesis	Gold	Pred.
The man was born in 1950.	The man was 18 in 1968.	E	C
No arrow hit the target.	Not all arrows hit the target.	C	E
Bob believes that Twin Peaks is the best tv show ever.	Twin Peaks is the best tv show ever.	N	E
All seagulls fly.	All birds fly.	N	E

Table 4: Error examples. The column Gold contains the correct tags, while the column Predicted contains the incorrect tags predicted by RoBERTa.

Label	Errors	Tot	%
Negation	116	568	62
Implicit Negation	30	146	16
Contrastive Negation	19	179	10
Partial Negation	16	32	8
Affixal Negation	5	75	3

Table 5: RoBERTa errors on negation relations.

In Table 5 it is evident that the largest number of errors belongs to the *Negation* macro-category. In this case, the sentences contained elements such as quantifiers, modals, temporal adverbs and relative pronouns. Therefore, it appears that the comprehension of negation is more difficult when it is related to these elements.

Label	Errors	Tot	%
Downward Monotone	189	222	48
Upward Monotone	25	138	6
Non-Monotone	62	98	16

Table 6: RoBERTa errors on hierarchical relations.

In Table 6 we can see the errors made by the RoBERTa in dealing with hierarchical relationships. Most errors relate to *Downward Monotone* and *Non-Monotone* sentences.

Label	Errors	Tot	%
Temporal Relation	64	182	19.94
Preconditions	53	146	16.51
World Knowledge	26	60	8.10
Spatial Relation	45	148	14.02
Cause/Effect	24	74	7.48

Table 7: RoBERTa errors on common-sense relations.

In Table 7 we show only the most relevant categories for what concerns the errors committed by the model dealing with *common-sense* and *common-knowledge*.

As we can see, *Temporal Relation*, *Preconditions* and *Spatial Relation* are the most difficult categories for the model to label correctly.

As illustrative examples, in Table 4 are four sentences mislabelled by RoBERTa. We note that the sentences are very simple and easy for human beings to understand.

6 Conclusions

Following a large number of recent studies (Naik et al., 2018), (Glockner et al., 2018), (Belinkov et al., 2019), (Liu et al., 2019a), we also tried to investigate whether the latest neural models were able to understand certain linguistic phenomena. On the one hand, we wanted to test the models on the real understanding of the English language, on the other hand, we wanted to build a fine-grained dataset, which allows a detailed analysis of each phenomenon. We tested two of the the most high-performance models such as BERT and RoBERTa and we observed how they struggle dealing with linguistic features that are quite simple to understand for a human being.

We have shown how the models can better handle phenomena such as *factivity* and *negation* if compared with the results obtained on *hierarchical relation* and *common-sense knowledge*. More in particular, we were able to stress how the state-of-the-art models struggle in dealing with linguistic phenomena that are essential for a correct understanding of the language such as the *possibility* generated by a statement, *temporal relations* between entities, the *negation* when there is a presence of *temporal adverbs* and *relative pronouns* and cases of *downward monotone* sentences. In future developments of our work we could use FANCY in order to perform fine tuning on Transformer-based models with the aim of increasing model performance and inferential capabilities. To do this it would be useful to produce more data, possibly annotated by different people, to test the models developed on different types of natural language. At the same time, the dataset could be implemented with other languages, such as Italian.

References

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Trans. Assoc. Comput. Linguist.*, 7:49–72.

Yonatan Belinkov, Adam Poliak, Stuart Shieber, Benjamin Durme, and Alexander Rush. 2019. On adversarial removal of hypothesis-only bias in natural language inference. pages 256–262.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642. Association for Computational Linguistics.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136. Association for Computational Linguistics.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Richard Feldman. 2003. *Epistemology*. Prentice Hall.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking NLI systems with sentences that require simple lexical inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics.

Laurence Horn. 1989. *A Natural History of Negation*. University of Chicago Press.

Ray S. Jackendoff. 1969. An interpretive theory of negation. *Foundations of Language*, 5(2):218–241. Publisher: Springer.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors.

Quoc V Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents.

Nelson F. Liu, Roy Schwartz, and Noah A. Smith. 2019a. Inoculation by fine-tuning: A method for analyzing challenge datasets. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv:1907.11692 [cs]*.

Bill MacCartney and Christopher D. Manning. 2009. An extended model of natural logic. In *Proceedings of the Eight International Conference on Computational Semantics*, pages 140–156, Tilburg, The Netherlands. Association for Computational Linguistics.

Bill MacCartney and Christopher D. Manning. 2014. Natural logic and natural language inference. In *Computing Meaning*, volume 47, pages 129–147. Springer Netherlands.

James D. McCawley. 1991. *The Syntactic Phenomena of English*.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

James Pustejovsky and Olga Batiukova. 2019. *The Lexicon*. Cambridge University Press, Cambridge, England.

Stephen Creig Roller. 2017. Identifying lexical relationships and entailments with distributional semantics.

Roser Saurí and James Pustejovsky. 2012. Are you sure that this happened? assessing the factuality degree of events in text. *Computational Linguistics - COLI*, 38:1–39.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in neural information processing systems*, volume 32. Curran Associates, Inc.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Automatic Assessment of English CEFR Levels Using BERT Embeddings

Veronica Juliana Schmalz[1,3], **Alessio Brutti**[1,2]
1. Free University of Bozen-Bolzano, Bolzano, Italy
2. Fondazione Bruno Kessler, Trento, Italy
3. KU Leuven, *imec* research group *itec*, Kortrijk, Belgium
veronicajuliana.schmalz@kuleuven.be, brutti@fbk.it

Abstract

The automatic assessment of language learners' competences represents an increasingly promising task thanks to recent developments in NLP and deep learning technologies. In this paper, we propose the use of neural models for classifying English written exams into one of the Common European Framework of Reference for Languages (CEFR) competence levels. We employ pre-trained Bidirectional Encoder Representations from Transformers (BERT) models which provide efficient and rapid language processing on account of attention-based mechanisms and the capacity of capturing long-range sequence features. In particular, we investigate on augmenting the original learner's text with corrections provided by an automatic tool or by human evaluators. We consider different architectures where the texts and corrections are combined at an early stage, via concatenation before the BERT network, or as late fusion of the BERT embeddings. The proposed approach is evaluated on two open-source datasets: the English First Cambridge open language Database (EFCAMDAT) and the Cambridge Learner Corpus for the First Certificate in English (CLC-FCE). The experimental results show that the proposed approach can predict the learner's competence level with remarkably high accuracy, in particular when large labelled corpora are available. In addition, we observed that augmenting the input text with corrections provides further improvement in the automatic language assessment task.

1 Introduction

Finding a system which objectively evaluates language learners' competences is a daunting task. Several aspects need to be considered, including both subjective factors, like age, native language, cognitive capacities of the learner, and learning-related factors, for example the amount and type of received linguistic input (James, 2005; Chapelle and Voss, 2008; Jang, 2017). Indeed, language competences are not holistic, but concern different domains, so that considering the mere formal correctness of learners' language has been shown not to represent a proper assessment procedure (Roever and McNamara, 2006; Harding and McNamara, 2017; Chapelle, 2017). Moreover, human evaluators, despite having to adhere to a predefined scale and guidelines, such as the CEFR (Council of Europe, 2001), have proved to be biased (Karami, 2013) and inaccurate (Figueras, 2012). For these reasons, new language testing methods and tools have been developed. Current state-of-the-art models, such as Transformers, allow to process numerous and complex linguistic data efficiently and rapidly, by means of attention-based mechanisms and deep neural networks that capture the relevant features for the targeted task. However, the creation and access to necessary language examination resources including annotations and metadata appear to date limited. In this paper, we propose using a series of BERT-base models to automatically assign CEFR levels to language learners' exams.

Our aim is examining the possibility of providing the system with previously generated corrections, either by humans or automatically with a language checker. Additionally, we want to analyse the impact of the amount of data on the accuracy of the model in the classification of written exams taken from the English First Cambridge Open Language Database (EFCAMDAT)

(Geertzen et al., 2013) and the Cambridge Learner Corpus for the First Certificate in English (CLC-FCE) (Yannakoudakis et al., 2011). In this way, a significant turning point could be made both in improving the functioning of these automatic systems and in the future collection of data from other languages.

2 Related Works

Automatic language assessment methods concern the creation of fast, effective, unbiased and cross-linguistically valid systems that can both simplify assessment and render it objective. However, achieving such results represents a complex task that researchers have been addressing for years while experimenting with several methodologies and techniques. The first developed tools used to mainly deal with written texts and exploited Parts-of-Speech (PoS) tagging to grade students' essays (Burstein et al., 2013), and latent semantic analysis to evaluate the content, providing also short feedback (Landauer, 2003). Advances in AI, NLP and Automatic Speech Recognition (ASR) led to the additional emergence of systems that assess spoken language skills, such as the *SpeechRater* (Xi et al., 2008), which considers clarity of expression, pronunciation and fluency. To date, several other automatic language assessment tools are applied in the domain of large scale testing, for example *Criterion* (Attali, 2004), *Project Essay Grade* (Wilson and Roscoe, 2020), *MyAccess!* (Chen and Cheng, 2008) and *Pigai* (Zhu, 2019). The first can detect grammatical and usage-based errors, as well as punctuation mistakes, providing also feedback. However, it requires being trained on the specific topics to assess. The second system exploits a training set of human-scored essays to score unseen texts, evaluating diction, grammar and complexity from statistical and linguistic models. Similarly, *MyAccess!*, calibrated with a large number of essays, can score learners' texts and measure advanced features such as syntactic and lexical complexity, content development and word choice, providing detailed feedback. On the contrary, *Pigai*, exploits NLP to compare the essays submitted by students with those contained in its corpora, measuring the distance between the two (Zhu, 2019). Despite the extreme efficiency of these tools, to perform accurately they generally need large amounts of labelled and human-corrected training data. Furthermore, a standard scale is needed, which can be extended between different groups of learners. In addition, powerful computational resources, and in certain cases, significant memory, are required. All these elements together constitute fundamental pre-requisites which can be difficultly fulfilled. For this reason, we present a distinct approach to the previous ones which, starting from different amounts of students' original texts, provides a classification within the different CEFR levels exploiting BERT-base models and subsidiary corrections.

3 Proposed Approach

The approach we propose for the automatic assessment of the language competences of adult English language learners is based on the use of Transformer-type architectures performing multi-class classification. Among these, BERT-based models, characterised by efficient parallel training and the capacity of capturing long-range sequence features, distinguish themselves for their size and amount of training data (Vaswani et al., 2017). Being pre-trained on generic large corpora, with Masked Language Modelling (MLM) and Next Sentence Prediction (NSP) strategies, they can be conveniently employed in a wide range of tasks, including text classification, language understanding and machine translation.

The models we use for our experiments are grounded on the *BERT-base-uncased* architecture, part of the Hugging Face Transformers Library released in 2019 (Wolf et al., 2020) and inspired by BERT (Devlin et al., 2018) from Google Research, that encodes input texts into low-dimensional embeddings. Our baseline model maps these compact representations into the CEFR levels using a network with two fully connected layers. Fig. 1(a) graphically represents the architecture. Note that this approach requires training the final classifier only. Retraining or fine-tuning the BERT model would probably require very large datasets which are not always available for this task. In order to augment the input text with corrections (either automatic or human) we investigate two possible directions. The first one (Fig. 1(b)) concatenates the two texts and applies the pre-trained BERT model. The resulting embeddings are expected to encode the information related to both texts. Conversely, the second architecture extracts individual embeddings for the original texts and the corrected ones.

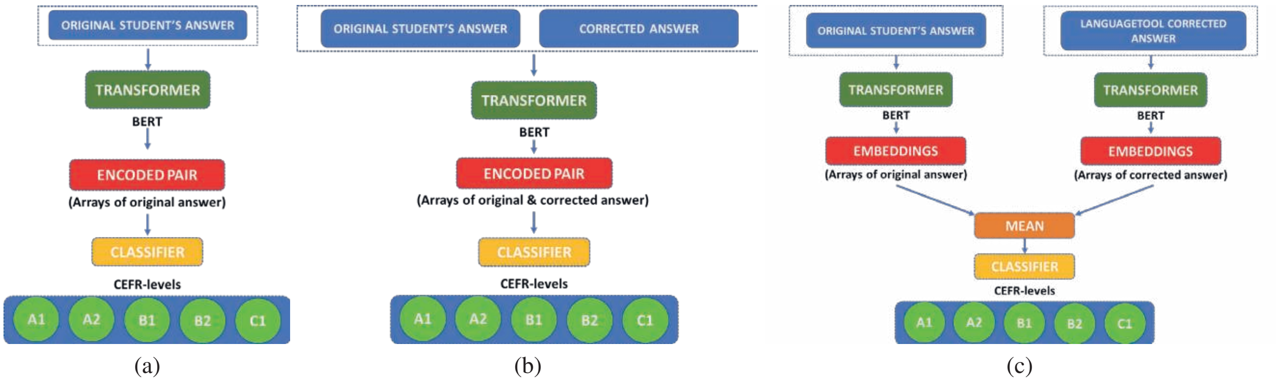


Figure 1: Proposed architectures for CEFR prediction. a) **Baseline**: original learners' texts as input; b) **Concatenation**: model taking the original learners' texts and the corrections concatenated; c) **Two-streams**: model processing the original learners' texts and the corrections with separate streams.

These are then merged and processed by the classifier, as shown in Fig. 1(c).

We resort to these types of models to be able to efficiently process texts capturing long-range sequence features thanks to parallel word-processing and self-attention mechanisms. Regardless of the length of the texts, the architecture should be, indeed, able to accurately categorise the examinations according to the CEFR A1, A2, B1, B2 and C1 levels of competence. These, in fact, are fed to the model as labels during the training together with single contextual embeddings, or concatenated ones if corrections are included. Note that we do not provide the model with any indication about the types of errors in the original text. This information is directly extracted by the model when processing the original text together with its corrected version.

4 Experimental Analysis

We evaluate the architectures described above, using both automatic and human corrections, on two English open-source datasets: EFCAMDAT and CLC-FCE. We also experiment varying the amount of training material. The performance of the models is measured in terms of weighted classification accuracy.

4.1 EFCAMDAT Dataset

The EFCAMDAT dataset constitutes one of the largest language learners datasets currently available (Geertzen et al., 2013). The version we use contains 1,180,310 essays submitted by adult English learners from more than 172 different nationalities, covering 16 distinct levels compliant with the CEFR proficiency ones. Each essay has been corrected and evaluated by language instructors; in addition to the original texts, their corrected versions and annotated errors are also included.

We considered a sub-set of the dataset comprising 100,000 tests. Table 1 reports the distribution of the exams across the different CEFR levels, including also the average numbers of violations identified by both humans evaluators and the automatic tool, normalized by the average text length. Note that the average errors per word decrease as the level of competence increases. Observe also that the automatic errors tend to be more numerous than the human ones, in particular for low competence levels. We use the official test partition composed of 1,447 essays. The development set is a 20% subset of the training set.

4.2 CLC-FCE Dataset

The CLC-FCE dataset is a collection of texts produced by adult learners for English as a Second or Other Language (ESOL) examinations from the First Certificate in English (FCE) written exam to attest a B2 CEFR level (Yannakoudakis et al., 2011). The learners' productions, consisting of two texts, have been evaluated with a score between 0 and 5.3 and the errors have been classified in 77 classes. Following the guidelines of the authors, the average score of the two texts has been mapped to CEFR levels, as shown in Table 2. Note that only 4 levels are available in this dataset and that the labels do not uniformly match the ones present in EFCAMDAT. Table 2 reports also the distributions of the texts across the 4 classes with the error partitions. We notice that, in this case,

levels	n. exams	average length	manual errors per word	automatic errors per word
A1	37,290	40	$4 \cdot 10^{-2}$	$10 \cdot 10^{-2}$
A2	36,618	67	$4 \cdot 10^{-2}$	$6 \cdot 10^{-2}$
B1	18,119	92	$4 \cdot 10^{-2}$	$5 \cdot 10^{-2}$
B2	6,042	129	$3 \cdot 10^{-2}$	$4 \cdot 10^{-2}$
C1	1,732	170	$2 \cdot 10^{-2}$	$3 \cdot 10^{-2}$

Table 1: EFCAMDAT dataset (sample of 100,000 exams): number of exams per CEFR level, mean text length (in tokens), mean number of manually and automatically annotated errors per word.

scores	levels	N. exams	average length	manual errors per word	automatic errors per word
0.0 - 1.1	A2	10	220	$16 \cdot 10^{-2}$	$7 \cdot 10^{-2}$
1.2 - 2.3	B1	417	205	$14 \cdot 10^{-2}$	$7 \cdot 10^{-2}$
3.1 - 4.3	B2	1,414	212	$9 \cdot 10^{-2}$	$6 \cdot 10^{-2}$
5.1 - 5.3	C1	265	234	$6 \cdot 10^{-2}$	$4 \cdot 10^{-2}$

Table 2: CLC-FCE dataset: assigned scores and number of exams per CEFR level, mean text length (in tokens), mean number of manually and automatically annotated errors per word.

manual errors have been annotated more in detail and they are indeed more numerous than the automatic ones. In general, the number of errors is higher than what observed in EFCAMDAT. Also for this corpus the average amount of errors per word, both automatic and manual, decreases as the level increases. The total number of texts within the corpus is 2,469. We employed a data partition according to which 2,017 examinations constituted the training set, whereas the remaining 194 constituted the test set. Differently, 10% of the training material represented the validation set. From the entire corpus we had to exclude 10 texts since they were not provided with an assigned score. Despite its small size, CLC-FCE represents an important resource given its systematic analysis of errors and the human corrections provided.

4.3 LanguageTool

In both datasets, the content written by language learners varies according to the levels of competence they were supposed to demonstrate. In addition to the human corrections provided with the data, we have generated automatic corrections using LanguageTool (Miłkowski, 2010), a language checker capable of detecting grammatical, syntactical, orthographic and stylistic errors to automatically correct texts of different nature and length (Naber and others, 2003). The automatic checker is based on surface text processing, does not use a deep parser and does not require a fully formalised grammar. By means of this, we have applied the pre-defined rules for the English language to the learners' essays, generating new correct texts for EFCAMDAT and for CLC-FCE. These were used as additional input data for the experiments.

4.4 Implementation Details

Our models have been implemented using Keras and Hugging-Face's pre-trained *BERT-base-uncased* architecture (Wolf et al., 2020). The models' encoder module, consisting of a Multi-Head Attention and Feed Forward component, receives as inputs the original learners' exams, together with additional possible human or automatic corrections. The transformed contextual embeddings are obtained applying *Global Average Pooling* to the outputs of the pre-trained frozen BERT Head. The classifier consists of a Dense layer of 768 units, with activation function *ReLu* and a Dropout rate of 0.2, followed by another Dense layer with less units, 128, and the same activation function and Dropout rate[1].

Lastly, the output layer consists of a Dense layer with *Softmax* as activation function and the models' final logits correspond to the different CEFR levels within which the texts are respectively clas-

[1]https://www.kaggle.com/akensert/bert-base-tf2-0-now-huggingface-transformer

N. Exams	text only	concatenation		two-streams	
		manual	automatic	manual	automatic
10K	95.2%	95.0%	**95.4%**	94.3%	94.4%
50K	97.1%	97.1%	97.0%	97.1%	97.0%
100K	97.4%	**97.7%**	97.3%	97.4%	97.2%

Table 3: Classification accuracy on EFCAMDAT using different amounts of training data, different inputs and different architectures.

sified. The selected loss is the *Sparse Categorical Cross-entropy* and the evaluation metric is the *accuracy*. The model is trained using *Adam* as optimizer with learning rate 10^{-5} for EFCAMDAT and 10^{-4} for CLC-FCE. The batch size is 32 and the input text maximum length is set to 450 for EFCAMDAT and 512 for CLC-FCE. These hyperparameters were optimized on the related development sets.

5 Experimental Results

Table 3 reports the classification accuracy on the EFCAMDAT test set using the proposed architectures in Fig. 1. Note that although EFCAMDAT features more than 1 million samples, we limit our analysis to 100K texts, due to memory issues and performance saturation. The results include also variations in the amount of training material, considering 10K and 50K training exams. These subsets have been obtained sampling in a uniform way the training set, therefore the distribution of exams per class does not change.

First of all, it is worth noting that the best approach reaches an extremely high classification accuracy (almost 98%). In addition, performance almost saturates with 50K essays, while with only 10K training samples the accuracy is well above 95%. The use of corrections, concatenated with the original text, provides some improvements over the model with original texts only. Automatic corrections seem to be more effective with less training data, while manual annotations outperform the baseline with larger training sets. The latter can, indeed, be more accurate, in particular for high proficiency levels, but their inherited variability makes the learning task more difficult. As a consequence, more training samples are needed to properly learn how to classify the input text. This is evident in Table 3 where the manual corrections are the worst for 10K samples, aligned with the baseline with 50K training samples, and the best performing when the 100K training texts are used. Finally, the two-stream approach averaging the BERT embeddings of the two texts, seems to be less performing, although by a small margin. Probably, the averaging operation does not represent the most suitable one in this context as it tends to generate embedding representations which are somehow intermediate between those of the original texts and those of the corrections and, hence, less discriminative.

Table 4 reports the results obtained on the CLC-FCE corpus. With respect to EFCAMDAT, this corpus is characterized by a smaller amount of training material and by a less consistent evaluation of the input text. These two facts lead to a clear reduction of the classification accuracy, as reported in the table. Due to the lower accuracy and smaller size of the training set, the final performance of each model has a certain degree of variability, which dependents on the model initialization and on the other random number generations in the training process. Therefore, we performed several runs varying the seed of the random number generator. The average accuracy, as well as the standard deviation, are also reported in Table 4.

model	accuracy
text only	61.5% $\pm$ 2.0
manual corr.	60.7% $\pm$ 1.8
autom. corr.	61.7% $\pm$ 1.8
two-streams	61.5% $\pm$ 1.3

Table 4: Classification accuracy on CLC-FCE using different architectures and types of corrections. The two-streams model uses automatic corrections. Results are averaged over multiple runs.

Given the limited size of the training set, it is not surprising to find rather similar results across all the models. As expected, the manual corrections are the worst performing, since they would require large training sets to learn how to handle human evaluations. It is worth pointing out that the amount of errors per word in CLC-FCE is much larger than in EFCAMDAT, which makes the learning task even more complex. Nevertheless, considering also the standard deviations, the models based on automatic corrections are slightly better than the model using the original texts only. The two-streams model appears extremely close to the concatenation model, but this could be related to the fact that the overall accuracy is not that high.

6 Conclusions

In this paper we presented an alternative approach for the efficient and unbiased assessment of the competences of English language learners using pre-trained BERT-base models. We structured a multi-class classification task to map the BERT embeddings of written exams from the EFCAMDAT and CLC-FCE open-source corpora to five different levels of the CEFR scale. Alongside the students' original texts and the provided manual corrections, we automatically generated additional corrected versions with LanguageTool, a multifaceted and versatile language checker . Thus, we conducted several experiments varying both the type and quantities of the models' input, as well as the typologies of models. Our results proved that BERT-based architectures remarkably succeed in classifying CEFR proficiency levels starting from original texts, especially with numerically significant data. Moreover, we observed that adding automatic and manual corrections can contribute to improve the quality of results.

References

Yigal Attali. 2004. Exploring the feedback and revision features of criterion. *Journal of Second Language Writing*, 14:191–205.

Jill Burstein, Joel Tetreault, and Nitin Madnani. 2013. The e-rater® automated essay scoring system. In *Handbook of automated essay evaluation*, pages 77–89. Routledge.

Carol A Chapelle and Erik Voss. 2008. Utilizing technology in language assessment. *Encyclopedia of language and education*, 7:123–134.

Carol A Chapelle. 2017. Evaluation of technology and language learning. *The handbook of technology and second language teaching and learning*, pages 378–392.

Chi-Fen Emily Chen and Wei-Yuan Eugene Cheng Cheng. 2008. Beyond the design of automated writing evaluation: Pedagogical practices and perceived learning effectiveness in efl writing classes. *Language Learning & Technology*, 12(2):94–112.

Education Committee Council of Europe, Council for Cultural Co-operation. 2001. *Common European Framework of Reference for Languages: learning, teaching, assessment*. Cambridge University Press.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Neus Figueras. 2012. The impact of the cefr. *ELT journal*, 66(4):477–485.

Jeroen Geertzen, Theodora Alexopoulou, Anna Korhonen, et al. 2013. Automatic linguistic annotation of large scale l2 databases: The ef-cambridge open language database (efcamdat). In *Proceedings of the 31st Second Language Research Forum. Somerville, MA: Cascadilla Proceedings Project*, pages 240–254. Citeseer.

Luke William Harding and Tim McNamara. 2017. Language assessment: The challenge of elf. In *Routledge Handbook of English as a Lingua Franca.* Routledge.

Carl James. 2005. Contrastive analysis and the language learner. *Linguistics, language teaching and language learning*, 120.

Eunice Eunhee Jang. 2017. Cognitive aspects of language assessment. *Language Testing and Assessment,*, pages 163–177.

Hossein Karami. 2013. The quest for fairness in language testing. *Educational Research and Evaluation*, 19(2-3):158–169.

Thomas K Landauer. 2003. Automatic essay assessment. *Assessment in education: Principles, policy & practice*, 10(3):295–308.

Marcin Miłkowski. 2010. Developing an open-source, rule-based proofreading tool. *Software: Practice and Experience*, 40(7):543–566.

Daniel Naber et al. 2003. A rule-based style and grammar checker.

Carsten Roever and Tim McNamara. 2006. Language testing: The social dimension. *International Journal of Applied Linguistics*, 16(2):242–258.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Joshua Wilson and Rod D Roscoe. 2020. Automated writing evaluation and feedback: Multiple metrics of efficacy. *Journal of Educational Computing Research*, 58(1):87–125.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Xiaoming Xi, Derrick Higgins, Klaus Zechner, and David M Williamson. 2008. Automated scoring of spontaneous speech using speechratersm v1. 0. *ETS Research Report Series*, 2008(2):i–102.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading esol texts. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 180–189.

Wenxin Zhu. 2019. A study on the application of automated essay scoring in college english writing based on pigai. In *2019 5th International conference on social science and higher education (ICSSHE 2019)*, pages 451–454.

Introducing a Gold Standard Corpus from Young Multilinguals for the Evaluation of Automatic UD-PoS Taggers for Italian

Veronica Juliana Schmalz[1,4], Jennifer-Carmen Frey[2], Egon W. Stemle [2,3]
1. Free University of Bozen-Bolzano, Bozen, Italy
2. Institute of Applied Linguistics, Eurac Research, Bozen, Italy
3. Faculty of Informatics, Masaryk University, Brno, Czech Republic
4. KU Leuven, *imec* research group *itec*, Kortrijk, Belgium
veronicajuliana.schmalz@kuleuven.be,
jennifercarmen.frey@eurac.edu, egon.stemle@eurac.edu

Abstract

Part-of-speech (PoS) tagging constitutes a common task in Natural Language Processing (NLP) given its widespread applicability. However, with the advance of new information technologies and language variation, the contents and methods for PoS-tagging have changed. The majority of Italian existing data for this task originate from standard texts, where language use is far from multifaceted informal real-life situations. Automatic PoS-tagging models trained with such data do not perform reliably on non-standard language, like social media content or language learners' texts. Our aim is to provide additional training and evaluation data from language learners tagged in Universal Dependencies (UD), as well as testing current automatic PoS-tagging systems and evaluating their performance on such data. We use Italian texts from a multilingual corpus of young language learners, LEONIDE, to create a tagged gold standard for evaluating UD PoS-tagging performance on non-standard language. With the 3.7 version of Stanza, a Python NLP package, we apply available automatic PoS-taggers, namely ISDT, ParTUT, POSTWITA, TWITTIRÒ and VIT, trained with diversified data, on our dataset. Our results show that the above taggers, trained on non-standard data or multilingual treebanks, can achieve up to 95% of accuracy on young multilingual learner data, if combined.

1 Introduction

Part-of-Speech (PoS) tagging relates to the assignment of tags or labels to the words, punctuation marks and symbols of a text. It constitutes a basic task in NLP, with applications ranging from machine translation to speech recognition and beyond. PoS-tags usually correspond to the morphosyntactic word classes of a given language, i.e. nouns, verbs, conjunctions, etc. Since each language contains specific linguistic characteristics that distinguish itself from others, tagsets are usually language dependent. The first automatic tool for the assignment of PoS-tags in the Italian language was the TreeTagger built at the University of Stuttgart (Schmid, 1994) to perform lemmatization and PoS-tagging contemporarily. Another milestone in the history of Italian PoS-tagging is the so-called Baroni's TreeTagger tagset, released in 2003. It represents the initially most adopted tagset, containing no less than 50 labels, half exclusively dedicated to verbs (Baroni et al., 2004). Along with the latter, *Tanl* (Attardi and Simi, 2009) constitutes an additionally relevant and comprehensive tagset for Italian. It counts with numerous tags and includes morphological word features. Three subcategories with different numbers of elements can be found in it, namely 14 coarse-grained tags, 37 fine-grained tags and 336 morphed tags.

Originally, automatic tagging methods were mainly employed with standard texts, such as essays, literature, and newspaper articles (Del

Monte et al., 2007; Baroni et al., 2004). However, with the advent of new communication systems and the expansion of language studies to more informal and common areas, attention started to shift to non-standard texts. In this regard, in several of the EVALITA periodic evaluation campaings for Italian NLP and speech tools, PoS tagging non-standard language has been a topic of interest (cf. Tamburini, 2007; Attardi and Simi, 2009; Bosco et al., 2016, Bosco et al., 2020). These tasks proved that PoS-tagging still represents an unsolved issue when it comes to less widely used language from different domains. Therefore, more studies and investigations are needed on specific language varieties.

Learner corpora exhibit a number of characteristics that differentiate them from the rest. In particular, numerous code-switching and code-mixing phenomena are common among them, as well as the presence of orthographical, syntactic and/or grammatical errors (Di Novo et al., 2019). More in detail, our data exhibited some peculiarities, for example the co-presence of variants for concepts ("Franco viene a casa e vede che *fuocare*/*brenn*") or new words combining different languages and morphologies ("Se sarò un giocatore famoso *richerò money*"). Given these distinctive aspects, analysing them in the context of PoS-tagging can offer interesting insights from the point of view of both the conception of these systems and their linguistic implications.

The rest of the paper is organized as follows. Section 2 provides relevant details concerning the Universal Dependencies (UD), as well as available Italian treebanks and taggers[1]. A brief overview about the differences in tagging standard and non-standard texts is presented in Section 3. Section 4 describes the methods and metrics commonly used for the evaluation of automatic taggers. We outline the tools and methodologies used for our experiments in Section 5 and the gold standard in Section 6. Next, in Section 7, we report the obtained results and in the subsequent section, namely 8, we discuss our findings, consider possible future works and draw our final conclusions.

2 Universal Dependencies and Italian Treebanks

Over the years, alongside the different taggers and treebanks of each language, a new language-independent framework in PoS annotation has emerged, the Universal Dependencies (UD). UD is a cross-linguistic project with the aim of building common annotation frameworks for several world languages. Underlying the Universal Dependencies annotation scheme are universal Stanford dependencies (Marneffe et al., 2008), Google universal PoS-tags (Petrov et al., 2011) and the Interset interlingua for morphosyntactic tagsets (cf. McDonald et al., 2013). In particular, for the Italian language, the UD counts seven different Treebanks. These are VIT, or the Venice Italian Treebank (Delmonte et al., 2007), ISDT, Italian Stanford Dependency Treebank (Bosco et al., 2014), ParTUT, or the Parallel Text Universal Treebank (Sanguinetti et al. 2014), PoSTwita (Bosco et al., 2016), TWITTIRÒ (Cignarella et al., 2018), Valico-UD (Di Novo et al., 2019) and PUD, or the Parallel Universal Dependencies Treebank (Zeman et al., 2018). The UD universal Italian tagset counts a total of 17 different labels (Universal Dependencies, 2021).

3 Pos-Tagging Standard vs Nonstandard Language

Among the various available treebanks and taggers for Italian, most have been created using exclusively standard data, such as newspaper articles, non-fictional texts, talks and Wikipedia pages for training the models (as in the case of VIT, ISDT, ParTUT and PUD). However, recently more attention has been placed on the creation of linguistic resources for nonstandard language, as the quantity and dissemination of this type of content increases exponentially, so does the need for suitable tools for its analysis and exploitation. In this respect, PoSTwita (Bosco et al., 2016) and TWITTIRÒ (Cignarella et al., 2018) resorted to additional non-standard Italian linguistic data from Twitter, while Valico-UD (Di Novo et al., 2019) used texts from Italian learners for the creation of their treebanks. Some of the main reasons why the use of standard language data outweighs that of nonstandard data are

[1] In this paper we use this term to refer to the Stanza models trained with the different available Italian Treebanks.

difficulties concerning the automatic processing and annotation of such texts. This applies especially when seeing the considerable amount of variation they contain, not only in the language itself, but also in the usage domains and among the individual language users (cf. Plank, 2016 and Sanguinetti et al. 2020). As a matter of fact, some distinctive features of non-standard texts are the broad variation in the structure and punctuation of utterances, namely in the syntax, but also at lexical level due to the use of abbreviations, domain-specific symbols or incorrect derivational forms, as well as code-switching for learners' language. The latter are likely to lead to issues regarding both automatic language processing, such as tokenization and lemmatization, and PoS-tagging, especially in the case of non-suitable or incomplete standard treebanks. For these reasons, the creation of resources from non-standard texts, like social media users or language learners, is crucial.

4 Evaluation of Automatic PoS-Taggers

When it comes to evaluating the performance of a PoS-tagger, generally an annotated gold standard reference corpus is used. The latter requires a distribution of the particular linguistic phenomena that is representative of the PoS-tagger's target application. Additionally, since a PoS-tagger combines several functions, like tokenization, word/sentence segmentation, and PoS-tag disambiguation, one of these parts must be firstly chosen as the test object. After selecting the aspect under analysis, it is necessary to choose which metrics to use to compare the results. The metrics commonly adopted for the evaluation of the tags assigned to a linguistic corpus are accuracy, precision, recall, F1-scores and Cohen's *K* (cf. Arstein and Poesio, 2008). These metrics vary not only in terms of the aspects they measure but also according to the type of data that constitute the corpus and its size.

Although various available UD taggers for Italian exist, little is known about how these perform on non-standard data. Some evaluations have been done on user-generated texts in social media (Bosco et al. 2016; Cignarella et al. 2018) and recently also on spoken language (Bosco et al. 2020) and adult learners of Italian with English, French, German and Spanish as first languages (Di Novo et al. 2019). However, this is still a nascent process, and the number of studies and analysed varieties are limited. Therefore, a closer examination and evaluation of an automatic tagger on an additional non-standard resource from a different domain promises to enhance our knowledge about PoS-tagging.

5 Methodology

In this study, we evaluate automatic PoS-tagging on the LEONIDE corpus (Glaznieks et al., 2020) to investigate how existing tagging models trained with the already available Italian treebanks perform with data from young language learners.

Given the inaccessibility of an evaluation sample for UD PoS-tagging on Italian learner language, we built our own pre-tokenized gold standard sample (see Section 6). Once we had our gold standard, we created a processing pipeline to test available tagging models for Italian on our data. For this, we used Stanza, a Python natural language analysis package designed using the UD formalism, as it offered easy access to a number of pre-trained models for PoS-tagging UD in Italian. The following models have been used in our evaluation: ISDT, ParTUT, POSTWITA, TWITTIRÒ and VIT. In order to evaluate only the PoS-tag disambiguation step of the PoS-taggers, regardless from other steps such as tokenization, we tagged the pre-tokenized texts using Stanza but deactivated the tokenizer (*tokenize_pretokenized=True*) and selected a different model as parameter each time. With the results obtained from each model, we resorted to *sklearn.metrics.classification_report* and *sklearn.metrics.cohen_kappa_score* to evaluate the total number of tags assigned to the more than 7,000 gold standard tokens according to accuracy and Cohen's *K*. In this way, the use of the exact same tokens and comparison metrics would have allowed an equal and meaningful comparison.

We closely focused on the accuracy and Cohen's *K* values (cf. Artstein and Poesio, 2008) because the first allowed us to check the overall performance of the tagger as well as the results on each tag's class, and the second to evaluate the similarity between the gold-standard and the automatically assigned tags.

As the available models had been trained on different data, both in quantity and type compared to each other but also compared to our corpus, it was particularly interesting to consider how they would deal with the young language learner data at hand, but also which type of errors they would make. We thus investigate common misclassifications for taggers and human annotators, discussing possible improvements and considerations to bear in mind when using these automatic PoS-tagging systems. For the latter, we used confusion matrices, so that we could check the types of errors made, and which were the most correctly assigned tags out of the total.

6 Gold Standard

For the creation of our gold standard, we used a subset of the Longitudinal lEarner cOrpus iN Italian, Deutsch, English (LEONIDE) (Glaznieks et al., 2020), a collection of 2,512 texts from 163 trilingual pupils attending lower secondary school (*scuola media*) in South Tyrol. The corpus contains texts in three languages, namely English, German, and Italian, and in two text genres, meaning *narrative* in the form of a picture-inspired story and *argumentative* in the form of a simple opinion text. Over the span of three years, the pupils were asked to write one text for each of the three languages and each of the text genres per year. The portion of Italian data in the corpus amounts to 844 texts counting 93,378 tokens. For our gold standard[2], we randomly selected a sample of 10% of the total available Italian texts, i.e. 84 texts with 7,665 tokens. We pre-tokenized and pre-tagged the texts in the sample using Stanza with the combined PoS-tagging model[3] in order to present our annotators with vertical files with one token per line and a PoS-tag to be eventually corrected. Once this step was completed, two language experts, native speakers of Italian, independently annotated the texts, correcting and adjusting the automatically pre-tagged version using the guidelines and documentation for the UD PoS tags and making use of the whole UD tagset. Their inter-annotator agreement in the independent tagging was relatively high, achieving a Cohen's Kappa of 0.98. In order to investigate a possible effect given by the use of a pre-tagged corpus version by the annotators, we also tested tagging the texts from scratch, meaning without any pre-assigned labels in the tokenized texts. For this purpose, we selected a random sample of ten texts extracted from the original corpus. Once again, to compare the two tagged versions we calculated the Cohen's *K* value, which resulted in 0.95. Hence, we can conclude that the pre-tagged version had no particular effect on the annotators and did not significantly affect their annotation.

Despite the generally good agreement between the annotators, some difficulties emerged. These mainly concerned cases of German code-switching, particles, clitic pronouns and auxiliary verbs (see Discussion), and occasionally orthographical or overgeneralization errors (ex. Da grande *facherò* [X/VERB] il calciatore). For the gold standard these issues were unanimously resolved in accordance with the Italian UD guidelines[4].

7 Results

Table 1 displays the obtained results in terms of tagging models' accuracy and Cohen's *K*, this time comparing the gold standard and the taggers' assigned tags, along with the accuracy scores reported in Stanza for the CoNLL 2018 Shared Task[5] on UD v2.5 Treebanks evaluation.

Tagger	Training data (in tokens)	Accuracy (Stanza)	Accuracy on learner data	Cohens' K (learner data)
Combined	Pre-trained	-	**0.95**	0.94
TWITTIRÒ	28,387 (ironic tweets)	0.94	0.86	0.84
ParTUT	Multilingual parallel treebank	0.98	0.84	0.82
PoSTWITA	119,238 (Tweets)	0.96	0.79	0.77
ISDT	278,429 (articles, newspapers, legal texts, Wikipedia)	0.98	0.76	0.73
VIT	272,000 (news, bureaucracy, finance, science, literature texts)	0.95	0.75	0.72

Table 1. Comparison of taggers' results on the LEONIDE's dataset (with additional training information) in terms of accuracy and Cohens' K values.

The highest accuracy on our gold standard for learner data has been achieved by the combination of models chosen by Stanza per default. We

[2]Available at http://hdl.handle.net/20.500.12124/34

[3]This indicates the Stanza model which originates from a combination of the existing taggers given by the Treebanks for the Italian language https://stanfordnlp.github.io/stanza/combined_models.

[4]https://universaldependencies.org/it/

[5]https://universaldependencies.org/conll18/evaluation.html

would have expected better results from ISDT, considering the high accuracy values on the standard data used to train it, and PoSTWITA for non-standard texts. However, regardless of this, in respect to our gold standard, the best models for accuracy value and Cohen's K are TWITTIRÒ and ParTUT. These latter performed well despite the fact that their tagsets did not contain all the tags used in our gold standard. In fact, both TWITTIRÒ and ParTUT, as well as PoSTWITA, did not include the PART tag (contrary to the other treebanks such as VIT and ISDT) , and thus did not assign it to particles. However, our human annotators referred to this tag to mark the common use of pronominal, reflexive and adverbial particles, such as 'mi' and ‘si’ in the corpus (ex. Più lingue *ci* sono; *Si* deve studiare molto). Furthermore, the parTUT treebank also lacked the tag for interjections, INTJ, as opposed to other treebanks that did make use of this category. Nevertheless, the training data for TWITTIRÒ was the treebank provided in Stanza that was closest to our data in type. It was created using data from social networks, therefore far from the scientific, nonfictional, or journalistic canon. On the other hand, ParTUT had been designed using standard texts but in Italian, English, and French in parallel.

8 Discussion

The results show that the performance of the models was significantly influenced by the particular type of data in our gold standard corpus, which presented incorrect orthographical or morphological tokens, but also contained numerous foreign words and abnormally disposed parts-of-speech within the sentence.

In fact, when inspecting the tags incorrectly assigned by the different models with confusion matrices (see Figure 1, 2 and 3 below), we noticed that:

- The foreign or misspelled words, which according to the UD rules had to be assigned the X tag, proved to be those with the highest number of errors. In fact, they were often confused with proper nouns, PROPNs, especially in the case of code-switching with the German language, where nouns are spelled with initial capital letters (ex. Dopo la scuola media voglio fare la *Hotelfachschule* [~~PROPN~~-X]). This was particularly evident with the ParTUT model that did not assign the X tag at all (see Figure 3);
- The second most incorrectly tagged words were particles, PART, which are not included in the tagsets of all models although they could have been assigned to pronominal, reflexive and adverbial particles (see section 7). Instead, these words were usually assigned the PRON tag for pronouns (ex. *Si* [~~PRON~~-PART] deve parlare questa lingua);
- The third most inaccurate group of tagged words was that of interjections, INTJ, which were also not included in all treebanks. These were often confused with particles or foreign words, PART or X (ex. *Ehm* [~~X~~-INTJ] ciao! fece Alessandra) as it is visible from Figure 2 in the case of the TWITTIRÒ model.

On the other hand, regarding discrepancies between the tags assigned by the human annotators, we found that:

- The groups on which there was most disagreement between the two annotators concerned particles, PART, and auxiliary verbs, AUX[6] (ex. Le strategie che funzionano peggio *sono* [~~AUX~~-VERB] <u>studiare</u> con il computer). Concerning the first, the models did not always include the PART tag in their employed tagsets. Auxiliary verbs, additionally, were also at times abnormally positioned within the sentence and were often automatically annotated incorrectly.
- Foreign words were often not annotated according to the X tag, probably because the annotators also had knowledge of the German language and therefore tended to assign the corresponding tag in the other language (ex. Faccio la *Landesberufschule* [~~NOUN~~-X]).

We can therefore argue that there were errors common to both automatic models and annotators, although the reasons for the errors were evidently different.

[6] This might be due to the fact that annotators could be influenced by the presence next to each token of a tag automatically assigned by Stanza, that had performed the tokenization of the texts.

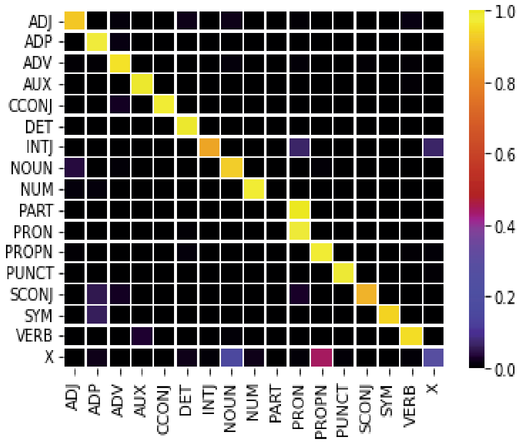


Figure 1. Confusion matrix related to the *combined* tagger (95% accurate)

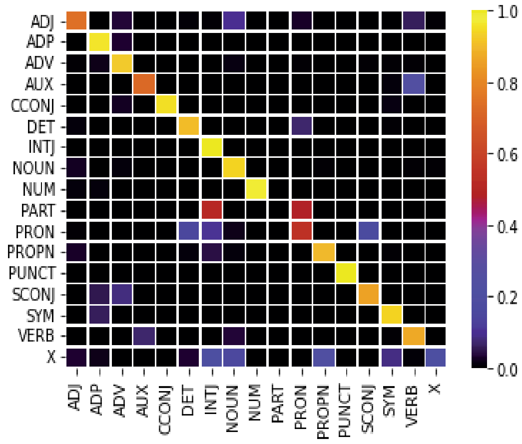


Figure 2. Confusion matrix related to the *TWITTIRÒ* tagger (86% accurate)

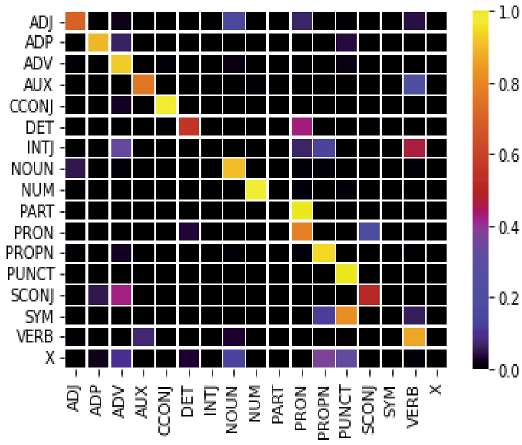


Figure 3. Confusion matrix related to the *ParTUT* tagger (84% accurate)

9 Conclusion

Although all taggers managed to execute the task of automatically PoS-tagging pre-tokenized Italian non-standard language with an accuracy of at least 75% (with the combined model offered by Stanza showing the best performance with 95% accuracy and 0.94 Cohen's *K*), there were differences in the performance shown by the individual models. The best performing two individual models were TWITTIRÒ (86%) and ParTUT (84%), while ISDT and PoSTWITA, that performed better in other evaluation tasks (Bosco et al. 2014, Cignarella et al. 2018) had a lower accuracy on our data. These results hint towards the fact that in order to automatically tag non-standard texts relating to language learners, the use of high-performance systems in the generic task is not sufficient, but the characteristics of the actual texts must also be taken into account.

Improvements could be made in the future regarding the adaptation of the models to the particular type of data used here. They could be, indeed, re-trained again in case a complete Treebank with Italian non-standard data becomes available. In addition, further attempts could be made to adapt or add the missing tags to the tagsets of all models so as not to have results biased by the lack of matching tags. Finally, as far as the annotators are concerned, they could be provided with the automatically pre-tokenized texts from the models, but in order to avoid pre-assigned tags influencing their annotation process, it would be preferable to omit these. Thus, human annotators would only get the taggers' tokenized text versions, so that the same tokens will be available for everyone, while the assignment of PoS would be completely up to them.

References

Artstein, Ron and Poesio, Massimo (2008). Inter-Coder Agreement for Computational Linguistics. Association for Computational Linguistics.

Attardi, Giuseppe and Simi, Maria (2009). Overview of the EVALITA 2009 Part-of-Speech tagging task. *Poster and Workshop proceedings of the 11th Conference of the Italian Association for Artificial Intelligence,* 12 December 2009, Reggio Emilia.

Baroni, Marco, Bernardini, Silvia, Comastri, Federica, Piccioni, Lorenzo, Volpi, Alessandra, Aston, Guy and Mazzoleni, Marco (2004). Introducing the La Repubblica Corpus: A Large, Annotated, TEI(XML)-Compliant Corpus of Newspaper Italian. *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04).* Lisbon: ELDA, 1771-1774.

Bosco, Cristina, Dell'Orletta, Felice, Montemagni, Simonetta, Sanguinetti, Marco and Simi, Maria (2014). The Evalita 2014 Dependency Parsing task. *Proceedings of the First Italian Conference on Computational Linguistics CLiC-it 2014 & and of* *the Fourth International Workshop EVALITA 2014*, 9-11 December 2014, Pisa, 1-8.

Bosco, Cristina, Tamburini, Fabio, Bolioli, Andrea and Mazzei, Alessandro (2016). Overview of the EVALITA 2016 Part Of Speech on TWitter for ITAlian task. In: Tamburini, F. (Ed.), *EVALITA Evaluation of NLP and Speech Tools for Italian. Proceedings of the Final Workshop,* Torino: Accademia University Press, 78-84.

Bosco, Cristina, Ballaré, Silvia, Cerruti, Massimo, Goria, Eugenio, & Caterina, Mauri (2020). KIPoS@ EVALITA2020: overview of the task on Kiparla part of speech tagging. In EVALITA 2020 Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (pp. 1-8). CEUR.

Cignarella, Alessandra Teresa, Bosco, Cristina, Patti, Viviana, & Lai, Mirko (2018). Application and analysis of a multi-layered scheme for irony on the Italian Twitter Corpus TWITTIRÒ. In: Calzolari, Nicoletta, Choukri, Khalid, Cieri, Christopher, Declerck, Thierry, Goggi, Sara, Hasida, Koiti, Isahara, Hitoshi, Maegaard, Bente, Mariani, Joseph, Mazo, Hélène, Moreno, Asuncion, Odijk, Jan, Piperidis, Stelios and Tokunaga, Takenobu (Eds.), *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018),* Miyazaki*:* European Language Resources Association, 4204-4211.

De Marneffe, Marie Catherine and Manning, Christopher D. (2008). Stanford typed dependencies manual. Technical report, Stanford University, 338-345.

Delmonte, Rodolfo, Bristot, Antonella and Tonelli, Sare (2007). VIT - Venice Italian Treebank: Syntactic and Quantitative Features. In: De Smedt, Koenraad, Hajic, Jan and Kübler, Sandra (Eds.), *Proceedings Sixth International Workshop on Treebanks and Linguistic Theories*, Bergen: Northern European Association for Language Technology (NEALT) Proceedings Series Vol.1, 43-54.

Di Novo, Elisa, Bosco, Cristina, Mazzei, Alessandro and Sanguinetti, Manuela (2019). Towards an Italian Learner Treebank in Universal Dependencies. In: Bernardi, Raffaella, Navigli Roberto and Semeraro, Giovanni (Eds.), *Proceedings of the 6th Italian Conference on Computational Linguistics, CliC-it 2019* (Vol. 2481), Bari, Italy, CEUR WS: 1-6.

Glaznieks, Aivars, Frey, Jennifer-Carmen, Stopfner, Maria, Zanasi, Lorenzo and Nicolas, Lionel (2020): LEONIDE: A longitudinal trilingual corpus of young learners of Italian, German and English. *International Journal of Learner Corpus Research* (IJLCR).

McDonald, Ryan, Nivre, Joakim, Quirmbach-Brundage, Yvonne, Goldberg, Yoav, Das, Dipanjan, Ganchev, Kuzman, Hall, Keith, Petrov, Slav, Zhang, Hao, Täckström, Oscar, Bedini, Claudia, Bertomeu Castelló, Nuria and Lee, Jungmee (2013). Universal Dependency Annotation for Multilingual Parsing. In: Schütze, Henrich, Fung, Pascale, Poesio, Massimo (Eds.), *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics,* Sofia: Association for Computational Linguistics*,* 92-97.

Petrov, Slav, Das, Dipanjan and McDonald, Ryan (2011). A universal part-of-speech tagset. In: Calzolari, Nicoletta, Choukri, Khalid, Declerk, Thierry, Uğur Doğan, Mehmet, Maegaard, Bente, Mariani, Joseph, Moreno, Asuncion, Odijk, Jan and Piperidis, Stelios (Eds.), *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), Instanbul: European Language Resources Association (ELRA),* 2089-2096.

Plank, Barbara (2016). What to do about non-standard (or non-canonical) language in NLP. In: Sharma Misra, Dipti, Sangal, Rajeev, Singh Kumar, Anil (Eds.), *Proceedings of the 13th Conference on Natural Language Processing*, Varanasi: NLP Association of India.

Sanguinetti, Manuela and Bosco, Cristina (2014). Converting the parallel treebank ParTUT in Universal Stanford Dependencies. In: Basili, Roberto, Lenci, Alessandro and Magnigni, Bernardo (Eds.), *Proceedings of the First Italian Conference on Computational Linguistics CLiC-it 2014 & and of the Fourth International Workshop EVALITA 2014: 9-11 December 2014, Pisa,* Pisa: Pisa University Press, 316-321.

Sanguinetti, Manuela, Cassidy, Lauren, Bosco, Cristina, Çetinoğlu, Özlem, Cignarella, Alessandra Teresa, Lynn, Teresa, Rehbein, Ines, Ruppenhofer, Josef, Seddah Djamé & Zeldes, Amir (2020). Treebanking user-generated content: a UD based overview of guidelines, corpora and unified recommendations. arXiv preprint arXiv:2011.02063.

Schmid, Helmut (1994): Probabilistic Part-of-Speech Tagging Using Decision Trees. *Proceedings of International Conference on New Methods in Language Processing*, Manchester, UK.

Tamburini, Fabio (2007). EVALITA 2007: The Part-of-Speech Tagging Task. *Contributi scientifici Associazione Italiana per l' Intelligenza Artificiale.* Anno IV, Giugno 2007.

Universal Dependencies (February 2021). UD for Italian. https://universaldependencies.org/it/.

Extracting Relations from Italian Wikipedia Using Self-Training

Lucia Siciliani[1], **Pierluigi Cassotti**[1], **Pierpaolo Basile**[1], **Marco de Gemmis**[1],
Pasquale Lops[1] **and Giovanni Semeraro**[1]
[1] Dipartimento di Informatica
Università degli Studi di Bari Aldo Moro, Bari, Italy
{firstname}.{surname}@uniba.it

Abstract

In this paper, we describe a supervised approach for extracting relations from Wikipedia. In particular, we exploit a self-training strategy for enriching a small number of manually labeled triples with new self-labeled examples. We integrate the supervised stage in WikiOIE, an existing framework for unsupervised extraction of relations from Wikipedia. We rely on WikiOIE and its unsupervised pipeline for extracting the initial set of unlabelled triples. An evaluation involving different algorithms and parameters proves that self-training helps to improve performance. Finally, we provide a dataset of about three million triples extracted from the Italian version of Wikipedia and perform a preliminary evaluation conducted on a sample dataset, obtaining promising results.

1 Introduction

The goal of an Open Information Extraction (Open IE) system is to extract relations occurring within a text written in natural language. Each relation is structured in the form of a triple that is composed by three elements i.e. $\{(arg1; rel; arg2)\}$. More specifically, given a relation, $arg1$ and $arg2$ can be nouns or phrases, while rel is a phrase that denotes the semantic relation between them. Open IE finds its application in several NLP tasks like Question Answering, Knowledge Graph Acquisition, Knowledge Graph Completion, and Text Summarization. For this reason, Open IE is gaining ever-growing attention as a research topic. Given the nature of the task, approaches for Open IE are deeply intertwined with the language of the corpora that have to be analyzed. Due to the availability of English corpora, the majority of the state-of-the-art works are specific for that language. For what concerns the Italian language, the model proposed by Guarasci et al. (2020) relies on verbal behavior patterns based upon Lexicon-Grammar features. In a previous work, we proposed WikiOIE (Cassotti et al., 2021), a framework in which Open IE methods for the Italian language can be easily developed with the aim of encouraging researchers to conduct further work also for under-represented languages. The first solutions developed in WikiOIE are unsupervised, relying merely on PoS tags patterns and dependency relations. In Cassotti et al. (2021) the triples extracted by WikiOIE underwent a deep error analysis. The error analysis reveals syntactic errors such as missing subject or incomplete object information and semantic errors such as generic subject or relation. In this work, we propose a supervised approach to automatically filter out non-relevant triples provided by WikiOIE and a self-training strategy. Self-training (Yarowsky, 1995) works iteratively: a classification model is trained on labeled data, the trained model is used to classify unlabeled data i.e. pseudo-labels, the classification model is retrained on labeled data and high-confident pseudo-labels. Specifically, we manually annotate a small number of triples extracted by WikiOIE. Afterward, the annotated triples are augmented using self-training. Finally, the set of triples obtained through self-training at the previous step is exploited to train a supervised model. The paper is structured as follows: after a brief introduction of state-of-the-art methods for Open IE, Section 3 provides details about the self-training and the supervised model behind our methodology. Section 4 reports the results of the evaluation, while Section 5 closes the paper.

2 Related Work

At first, the IE task was performed by extracting from the text relations that were defined a-priori. However, the increasing amount of corpora available nowadays makes this process unfeasible, thus creating the urge to propose novel solutions to tackle this problem.

The Open IE task was defined in 2008 by Etzioni et al. (2008). The three most important elements characterizing this task are the following: it is domain independent, meaning that the text relations must be extracted from, can be related to any topic, the extraction must be unsupervised, approaches to solve this task must take into account the amount of data available and must be scalable.

Along with the definition of a new task, the authors proposed a model called TextRunner. It applies an approach that is composed of three main modules. The first one is a *learner* that exploits a parser to label the training data as trustworthy or not and then uses the extracted information to train a Naive Bayes classifier. Next, the *extractor* uses POS-tag features to obtain a set of candidate tuples from the corpus, and only those labeled as trustworthy are kept. Finally, a module denominated *assessor* assigns a probability score to the tuples extracted at the previous step based on the number of occurrences in the corpus.

The learning-based approach used in TextRunner has also been applied by several other systems like WOE (Wu and Weld, 2010), OLLIE (Mausam et al., 2012), and ReNoun (Yahya et al., 2014). In particular, WOE exploits Wikipedia-based bootstrapping: the system extracts the sentences matching the attribute-value pairs available within the info-boxes of Wikipedia articles. This data is then used to build two versions of the system: the first one based on PoS-tags, regular expressions, and other shallow features of the sentence, the latter based on features of dependency-parse trees, thus obtaining better results than the other one but with a lack of performance in terms of speed.

In recent works, OIE has been treated as a sequence labeling task. In this setting, models are trained to extract triple elements, i.e., subject, predicate, and object using a modified BIO tag schema (Ratinov and Roth, 2009) that involves particular prefixes to represent the triple elements, i.e., A0, P, and A1. Hohenecker et al. (2020) provide an evaluation of different training strategies and different neural network architectures such as bidirectional Long short-term Memory (BiLSTM), Convolutional Neural Networks (CNNs), and Transformers improving the state-of-the-art on the OIE16 benchmark (Stanovsky and Dagan, 2016) which focuses on the English language.

3 Methodology

In this section, we describe our supervised approach based on self-training integrated into the information extraction system called WikiOIE[1]. Before discussing details about the supervised approach, it is necessary to recap how WikiOIE works. The input of the pipeline is represented by the textual format of the Wikipedia dump obtained through the WikiExtractor tool[2] (Attardi, 2015). The text is extracted from the Wikipedia dump and processed using the UDPipe tool (Straka and Straková, 2017). For this task, we use version 1 of UDPipe with version 2.5 of the *ISDT-Italian* model. We opt for UDPipe, since it is trained using Universal Dependencies data for over 100 languages. In this way, our system can be potentially used on different Wikipedia dumps of several languages. WikiOIE directly calls the REST API provided by UDPipe so that it is easy to change the endpoint and the model/language. Another advantage of using Universal Dependencies is the common tag-set that is defined for all the languages. PoS-tags[3] and syntactic dependencies[4] are annotated with shared sets of labels. Again, this feature also allows the system to be independent from the language. The Wikipedia dump is read line-by-line. Each line contains a fragment (passage) of text that is processed using UDPipe. The output of this process is a set of sentences, and each sentence is annotated with syntactic dependencies. The sentence is transformed into a dependency graph that is the input of the Wiki Extractor module. This module extracts facts from the sentence in the form of triples *(subject, predicate, object)* and assigns a score.

As aforementioned, each sentence occurring in the text is annotated by UDPipe that provides an-

[1] The code is available on GitHub: `https://github.com/pippokill/WikiOIE`.

[2] `https://github.com/attardi/wikiextractor/wiki/File-Format`

[3] `https://universaldependencies.org/u/pos/`

[4] `https://universaldependencies.org/u/dep/`

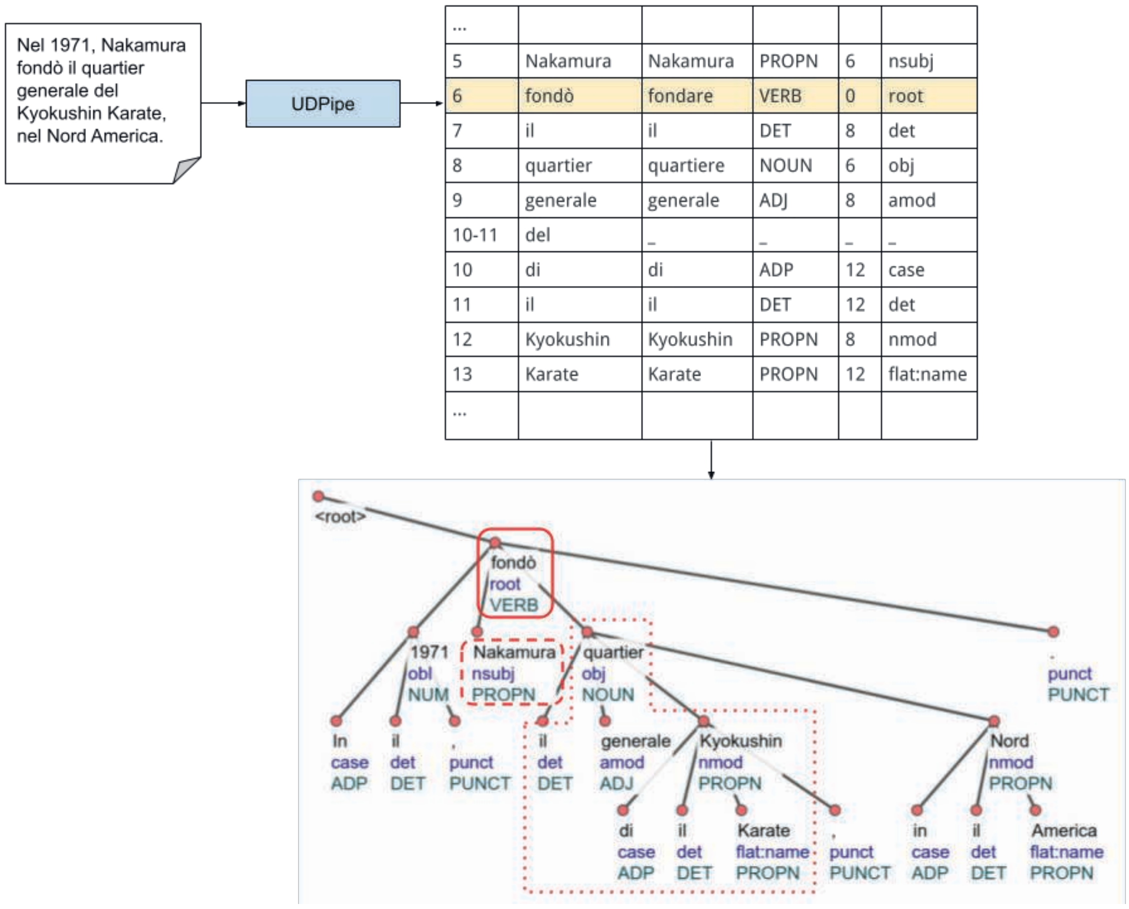


Figure 1: An example of UDPipe processing.

notations following the CoNLL-U format[5]. As shown in Figure 1, each token into the sentence is denoted by an index (first column) corresponding to the token position into the sentence (starting from 1). In the other columns are stored the features extracted by UDPipe, such as the token, the lemma, the universal PoS-tag, the head of the current word, and the universal dependency relation to the HEAD. If the head of the current word is equal to 0, it means that that token represents the head of the whole sentence, then the universal dependency relation will be equal to *root*. Figure 1 also reports the dependency graph of the sentence that is used by the Wiki Extractor module for extracting triples. We use an unsupervised pipeline based on both PoS-tag and dependencies to extract the first set of triples.

The first step of the extraction process consists of identifying sequences of PoS-tags that match verbs as reported in Table 1. In Table 1, the first column reports the PoS-tag patterns, while the second one reports an example of pattern usage. The sentence showed in Figure 1 matches the last pattern (VERB, *fondò*).

PoS-tag Pattern	**Example**
AUX VERB ADP	... *è nato nel* ...
AUX VERB	... *è nato* ...
AUX=(essere, to be)	... *è* ...
VERB ADP	... *nacque nel* ...
VERB	... *acquisì* ...

Table 1: Patterns of valid predicates.

When the information extraction algorithm finds a valid predicate pattern, it checks for a candidate subject and object for the predicate. A valid subject/object candidate must match the following constraints:

1. the candidate must be composed by a sequence of tokens belonging to the following PoS-tags: noun, adjective, number, determiner, adposition, proper noun;
2. the sequence of tokens composing the candi-

[5] `https://universaldependencies.org/format.html`

date can contain only one determiner and/or one adposition.

The candidate subject must precede the verb, while the candidate object must follow the predicate pattern. For the sentence in Figure 1 the candidate subject is "*Nakamura*", while the candidate object is "*il quartier generale di il Kyokushin Karate*"[6].

After identifying the candidate subject and object, the triple is accepted only if both the subject and the object have a syntactic relation with the verb. In particular, one of the tokens belonging to the subject/object must have a dependent relation with a token of the verb pattern.

More details about the unsupervised extraction of triples are reported in Cassotti et al. (2021).

3.1 Self-Training

Using the unsupervised approach, we obtain 3,562,803. We randomly select a subset of 200 triples for which the predicate occurs at least 20 times. Then, each triple is annotated by two experts as relevant (valid) or not-relevant. Details on this dataset and the results of the annotation process are reported in (Cassotti et al., 2021). For the self-training, we select only triples in which the two experts agree. Finally, we have a set of 137 triples that we call L.

From the whole set of 3.5M triples, we randomly select the 1% of unlabeled triples in which the predicate occurs at least 20 times. This subset is denoted as U. The set L is split in two subsets: L_t for training and L_v for validation. In particular, L_t is used as the initial dataset for the self-training procedure, while L_v is used for setting the initial parameters' values of the learning algorithm.

As a preliminary step, we search for the best parameters using L_t for training and L_v for validating the performance. We use the macro-averaged F1 score since our dataset is highly unbalanced: the 82% of the triples are labelled as relevant.

The self-training process works as follow:

1. from the set U, we randomly select p triples;
2. we train a supervised model using labeled triples in L_t;
3. the p triples are labeled using the trained model, and a confidence score is assigned to each classified triple;
4. the triples with a confidence score higher or equal to a threshold t are added to L_t by maintaining classes balance in L_t. If the classifier does not provide a confidence score, all instances labeled as valid are included in L_t;
5. if U contains at least p triples go to step 1 otherwise ends. The self-training loop can also be terminated if a specific number of iterations is reached.

The resulting set of labeled triples L_t is used to train the final model, which is employed to classify all the triples extracted using the unsupervised approach.

More details about both the parameters' values and the training algorithm are reported in Section 4.

3.2 Supervised Approach

For both the self-training and the classification of triples, we exploit algorithms provided by LibLinear[7]. In particular, we use both logistic regression and support vector classification: the former can provide a confidence score, while the latter cannot.

The set of features is selected by taking into account the supervised approaches already developed for English. In particular, we use:

- the PoS-tags occurring into the subject, object, and predicate;
- the sequence of PoS-tags that compose the predicate. This feature is also computed for both the subject and the object;
- the n-gram that composes the predicate;
- the set of dependencies that link the subject to the predicate;
- the set of dependencies that link the object to the predicate.

The C value of the learning algorithm is determined by performing a grid search using L_t for training and L_v for validating. Due to the small size of the original set L, we perform a 50/50 split. More details are reported in Section 4.

[6]It is important to note that UDPipe splits the articulated preposition "del" in "di:ADP" and "il:DET".

[7]https://www.csie.ntu.edu.tw/ cjlin/liblinear/

Method	**C**	P_0	R_0	$F1_0$	P_1	R_1	$F1_1$	**F1**
S_{log}	10	.54	.58	.56	.91	.89	.90	.62
S_{svc}	8	.60	.75	.66	.94	.89	.92	.73

Table 2: Results of the grid search.

Method	**Size**	P_0	R_0	$F1_0$	P_1	R_1	$F1_1$	**F1**	$\Delta\%$
S_{log}	15,771	.88	.58	.70	.92	.98	.95	.74	19.35%
S_{svc}	19,545	.60	.50	.55	.90	.93	.91	.60	-17.81%

Table 3: Results of the self-training approach.

4 Evaluation

The goal of the evaluation is twofold: 1) measure the performance and the contribution of the self-training; 2) evaluate the quality of the extracted triples. For the first goal, we evaluate how the new instances added to the initial set of training affect the performance. For the second goal, we manually annotated a small subset of extracted triples in order to evaluate their quality.

4.1 Evaluate Self-Training

The first step is to determine the best parameters for the learning algorithm. We use two algorithms: L2-regularized logistic regression (S_{log}) and L2-regularized L2-loss support vector classification (S_{svc}). For both algorithms, we perform a grid search for selecting the best value for the parameter C. Results of the grid search is reported in Table 2. In the table, we report the best C value for each approach. We denote with 0 the class of not-relevant triples, while 1 denotes relevant ones. F1 refers to macro-average F1. Results show that classifiers have poor performance in recognizing the class 0 since the dateset is both small and unbalanced.

We perform two self-training steps (one for each learning algorithm) using $p = 1,000$ and 20 as the number of maximum iterations. For the logistic regression, we set 0.85 as threshold. After the self-training step, we obtain a new training set which contains new instances. Table 3 reports for each learning approach the size of the new training set and the performance computed on the validation set. Moreover, the last column reports the increment of F1 with respect to the performance obtained before the self-training.

Experiments using self-training show that S_{log} is able to improve (+19%) its performance, while self-training has a negative impact on S_{svc} performance (-18%). Probably, this is due to the fact that it is not possible to set a threshold for selecting good classified instances during the self-training when the S_{svc} is involved. After observing the overall performances in both Tables 2 and 3, we select as training set for extracting triples the one obtained by S_{log} during the self-training. S_{log} is also able to both overcome the performance of S_{svc} obtained without self-training and achieve also an improvement in $F1_0$.

After the extraction and classification process, we obtain 2,974,374 triples[8] as reported in Table 4. The original set of triples extracted from the unsupervised approach was 3,562,803, this means that the 16.52% of unsupervised triples was classified as not-valid. Table 4 reports also information about the number of distinct subjects, objects, and predicates for both the unsupervised and supervised datasets. The supervised dataset is released in the same JSON format described in Cassotti et al. (2021).

4.2 Evaluate Triples

For the evaluation, we follow the same methodology proposed in Cassotti et al. (2021). In particular, we sample a subset of 200 triples from the final set of classified triples. The triples selected are the ones for which the predicate occurs at least 20 times. Then, each triple is annotated by two experts as relevant (valid) or not-relevant. We used Cohen's Kappa coefficient (K) to measure the pairwise agreement between the two experts. K is a more robust measure than simple percent agreement calculation since it takes into account the agreement occurring by chance. Higher values of K correspond to higher inter-rater reliability. Open

[8]The triples are available on Zenodo: `https://zenodo.org/record/5655028`. The triples obtained by the unsupervised approach are available here: `https://doi.org/10.5281/zenodo.5498034`.

Dataset	#triples	#dist. subj	#dist pred	#dist obj
unsupervised	3,562,803	1,298,481	269,551	2,030,742
supervised (S_{log})	2,974,374	1,189,648	241,053	1,720,348

Table 4: Dataset statistics.

Dataset	#valid (exp 1)	#ratio (exp 1)	#valid (exp 2)	#ratio (exp 2)	Kappa C.
unsupervised	115	0.64	161	0.81	0.24
supervised (S_{log})	158	0.79	163	0.82	0.63

Table 5: Results of the annotation process.

IE task lacks a formal definition of triple relevance thus for the annotation process, we adopt the concept of triple relevance reported in (Stanovsky and Dagan, 2016) that is based on assertiveness, minimalism, and completeness. This ensures that: the triples extracted still enclose the semantics of the original sentence (assertiveness), each element of the triple is as compact as possible without any unnecessary In our evaluation, we decide to give less weight to minimalism and focus more on the extraction completeness. After the annotation, we compute the ratio of relevant triples (column #ratio in Table 5) for each dataset and expert. Specifically, the ratio is computed dividing the number of triples annotated as relevant by the number of sampled triples.

Results of the evaluation are reported in Table 5, where also the previous results on the set of unsupervised triples is reported. It is important to highlight that the two datasets are not directly comparable since they are composed of different triples. In particular, a small subset of the unsupervised dataset is used to train the supervised one as explained in Section 3. Cohen's kappa coefficient for each dataset is provided in the last column of Table 5.

We obtain a good result in terms of number of valid triples. In particular, the supervised model provides a set of triples that improve the agreement between annotators. The supervised approach removes noisy and ambiguous triples since the initial subset L_t used for self-training contains only triples for which the annotators agree.

In this task, it is not always possible to compute standard metrics such as recall since it is not easy to determine the total number of valid triples due to the task's "open" nature. As future work, we plan to extend the number of manually annotated triples for performing a more rigorous evaluation and comparison of different information extraction methods for Italian.

5 Conclusions and Future Work

We propose a self-training strategy for implementing a supervised open information extraction system for the Italian version of Wikipedia. Our approach exploits a small set of manually labeled triples for expanding the training set. We integrate this system into WikiOIE, which is a framework for open information extraction on Wikipedia dumps. WikiOIE exploits UDPipe as a tool for processing and annotating the text and can be extended by adding several information extraction approaches.

We perform an extensive evaluation for measuring the impact of self-training on the overall classification performance. Results prove that self-training is able to improve the classification performance and help to identify not-relevant triples.

Finally, we sampled a subset of extracted triples, evaluated by two experts. The number of relevant triples increases when the self-training strategy is used by also improving the agreement between annotators.

As future work, we plan to extend the evaluation to a larger scale study, exploit several learning algorithms, and explore the application of the approach to other languages.

Acknowledgments

This research was partially funded by the INTERREG-MEDITERRANEAN Social and Creative project, priority axis 1: Promoting Mediterranean innovation capacities to develop smart and sustainable growth, Programme specific objective 1.1 To increase transnational activity of innovative clusters and networks of key sectors of the MED area (2019-2022).

References

Giusepppe Attardi. 2015. Wikiextractor. `https://github.com/attardi/wikiextractor`.

Pierluigi Cassotti, Lucia Siciliani, Pierpaolo Basile, Marco de Gemmis, and Pasquale Lops. 2021. Extracting Relations from Italian Wikipedia using Unsupervised Information Extraction. In Vito Walter Anelli, Tommaso Di Noia, Nicola Ferro, and Fedelucio Narducci, editors, *Proceedings of the 11th Italian Information Retrieval Workshop 2021 (IIR 2021)*. CEUR-WS. http://ceur-ws.org/Vol-2947/paper2.pdf.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. 2008. Open information extraction from the web. *Commun. ACM*, 51(12):68–74.

Raffaele Guarasci, Emanuele Damiano, Aniello Minutolo, Massimo Esposito, and Giuseppe De Pietro. 2020. Lexicon-Grammar based open information extraction from natural language sentences in Italian. *Expert Syst. Appl.*, 143.

Patrick Hohenecker, Frank Mtumbuka, Vid Kocijan, and Thomas Lukasiewicz. 2020. Systematic comparison of neural architectures and training approaches for open information extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8554–8565, Online, November. Association for Computational Linguistics.

Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. 2012. Open Language Learning for Information Extraction. In Jun'ichi Tsujii, James Henderson, and Marius Pasca, editors, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL*, pages 523–534, Jeju Island, Korea, 7. ACL.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 147–155, Boulder, Colorado, June. Association for Computational Linguistics.

Gabriel Stanovsky and Ido Dagan. 2016. Creating a Large Benchmark for Open Information Extraction. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016*, pages 2300–2305, Austin, Texas, USA, 11. The Association for Computational Linguistics.

Milan Straka and Jana Straková. 2017. Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe. In Jan Hajic and Dan Zeman, editors, *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada, 8. Association for Computational Linguistics.

Fei Wu and Daniel S. Weld. 2010. Open Information Extraction Using Wikipedia. In Jan Hajic, Sandra Carberry, and Stephen Clark, editors, *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 118–127, Uppsala, Sweden, 7. The Association for Computer Linguistics.

Mohamed Yahya, Steven Whang, Rahul Gupta, and Alon Y. Halevy. 2014. ReNoun: Fact Extraction for Nominal Attributes. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 325–335, Doha, Qatar, 10. ACL.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, Massachusetts, USA, June. Association for Computational Linguistics.

Sentiment Analysis of Latin Poetry: First Experiments on the Odes of Horace

Rachele Sprugnoli, Francesco Mambrini, Marco Passarotti, Giovanni Moretti
CIRCSE Research Centre, Università Cattolica del Sacro Cuore
Largo Agostino Gemelli 1, 20123 Milano
{rachele.sprugnoli,francesco.mambrini
marco.passarotti,giovanni.moretti}@unicatt.it

Abstract

In this paper we present a set of annotated data and the results of a number of unsupervised experiments for the analysis of sentiment in Latin poetry. More specifically, we describe a small gold standard made of eight poems by Horace, in which each sentence is labeled manually for the sentiment using a four-value classification (positive, negative, neutral and mixed). Then, we report on how this gold standard has been used to evaluate two automatic approaches for sentiment classification: one is lexicon-based and the other adopts a zero-shot transfer approach.[1]

1 Introduction

The task of automatically classifying a (piece of) text according to the sentiment conveyed by it, known as Sentiment Analysis (SA), is usually performed for purposes such as monitoring contents of social media or evaluating customer experience, by analysing texts like tweets, comments, and micro-blogs.

A still under-investigated yet promising research area where developing and applying SA resources and techniques is the study of literary texts written in historical and, particularly, Classical languages (e.g. Ancient Greek and Latin). Actually, investigating the lexical properties of Classical literary texts is a century-long common practice. However, such investigation can nowadays (1) lead to replicable results, (2) benefit from techniques developed for analysing the sentiment conveyed by any type of text and (3) be performed with freely available lexical and textual resources. As for the latter, the research area dedicated to building and using linguistic resources for Classical languages has seen a substantial growth during the last two decades (Sprugnoli and Passarotti, 2020). For what concerns SA, we recently built a polarity lexicon for Latin nouns and adjectives, called *LatinAffectus*. The current version of the lexicon includes 4,125 Latin lemmas with their corresponding prior polarity value (Sprugnoli et al., 2020b). *LatinAffectus* was developed in the context of the *LiLa: Linking Latin project* (2018-2023)[2] (Passarotti et al., 2020) which aims at building a Knowledge Base of linguistic resources for Latin based on the Linked Data paradigm, i.e. a collection of several data sets described using the same vocabulary of knowledge description and linked together. *LatinAffectus* is connected to the Knowledge Base, thus making it interoperable with the other linguistic resources linked so far to LiLa (Sprugnoli et al., 2020a).

In this paper we describe the use of *LatinAffectus* to perform SA of the *Odes* (*Carmina*) by Horace (65 - 8 BCE). Written between 35 and 13 BCE, the *Odes* are a collection of lyric poems in four books. Following the models of Greek lyrical poets like Alcaeus, Sappho, and Pindar, the *Odes* cover a wide range of topics related to the individual and social life in Rome during the age of Augustus, like love, friendship, religion, morality, patriotism, the uncertainty of life, the cultivation of tranquility and the observance of moderation. In spite of a rather lukewarm initial reception, the *Odes* quickly became a capital source of influence, in particular as a model of authorial voice and

[1]This paper is the result of the collaboration between the four authors. For the specific concerns of the Italian academic attribution system, Rachele Sprugnoli is responsible for Sections 2, 3, 4.2, 5; Marco Passarotti is responsible for Section 1; Francesco Mambrini is responsible for Section 4.1. Giovanni Moretti developed the zero-shot classification script.

[2]https://lila-erc.eu

identity.[3] Considering not only the importance of the *Odes* in the history of Latin and European literature, but also the diversity of the contents and tones of the poems collected therein, we argue that performing SA on such work can lead to interesting results and might represent a use case to open a discussion about the pros and cons of applying SA techniques and resources to literary texts written in ancient languages.

All data presented in this paper are publicly released: `https://github.com/CIRCSE/Latin_Sentiment_Analysis`.

2 Related Work

The majority of linguistic resources and applications in the field of SA involve non-literary and non-poetic texts, such as news and user-generated content on the web (Medhat et al., 2014). However, affective information plays a crucial role in literature and, in particular, in poetry where authors try to provoke an emotional response in the reader (Johnson-Laird and Oatley, 2016). Annotated corpora of poems and SA systems specifically designed for poetry are not as numerous as those in other areas of research, first of all that of social media, but works have been carried out for several languages,[4] including Arabic (Alsharif et al., 2013), Spanish (Barros et al., 2013), Odia (Mohanty et al., 2018), German (Haider et al., 2020), Classical Chinese (Hou and Frank, 2015) and, of course, English (Sheng and Uthus, 2020; Sreeja and Mahalakshmi, 2019).

Available annotated corpora of poems differ from each other from at least four points of view: annotation procedure (either involving experts or using crowdsourcing techniques), unit of analysis (verse, stanza, whole poem), granularity of classification (from binary classes, such as positive and negative, to wide sets of emotions), foci of the emotions (annotation of the emotions as depicted in the text by the author or as felt by the reader). With respect to previous work, in this paper we chose to involve experts, to perform annotation at the sentence level (as an intermediate degree of granularity between verse and stanza), to assign four generic classes without defining the specific emotion conveyed by the text, and to focus on the sentiment as depicted by the author.

As for automatic classification systems, the literature reports both lexicon-based (Bonta and Janardhan, 2019) and machine learning approaches, with a constant increasing use of deep learning techniques (Zhang et al., 2018). For example, Mohanty et al. (2018) experiment with Linear-SVM, Naive-Bayes and Logistic Regression classifiers on Odia poems, while Haider et al. (2020) perform multi-label classification on German stanzas with BERT. Given the lack of training data for Latin poetry, in this paper we will instead test unsupervised approaches.

3 Gold Standard Creation

3.1 Annotation

The Gold Standard (GS) consists of eight randomly selected odes,[5] two from each of the four books that make up the work, for a total of 955 tokens, without punctuation, and 44 sentences (average sentence length: 21, standard deviation: 11). Texts were taken from the corpus prepared by the LASLA laboratory in Liège.[6] We performed a single-label annotation of the original Latin text by Horace at sentence level. We have chosen the sentence as unit of annotation because it represents an intermediate degree of granularity between that of the verse and that of the stanza. In fact, the limited length of a verse can hinder the full understanding of the sentiment it conveys, while a stanza, being longer, risks to contain very different content and thus, potentially, even opposite sentiments. Furthermore, not all poems can be divided into stanzas, as this depends on the metric scheme of the poem. Instead, sentences can be detected in every poem regardless of its metric scheme, and represent a unit of meaning in their own right.

In the annotation phase, we involved two experts in Latin language and literature (A1 and A2) and another annotator with basic knowledge of Latin but provided with previous experience in sentiment annotation (A3). Annotators were asked to identify the sentiment conveyed by each sentence in the GS, taking into consideration both the vocabulary used by the author and the images that are evoked in the ode. More specifically, annotators were asked to answer the following question: which of the following classes best describes how

[3] For an orientation on the vast subject of the fortune and reception of the *Odes* see Baldo (2012).

[4] For a recent survey on sentiment and emotion analysis applied to literature, see Kim and Klinger (2018).

[5] Book I: odes 10 and 17; Book II: odes 7 and 13; Book III: odes 13 and 23; Book IV: odes 7 and 11.

[6] `http://web.philo.ulg.ac.be/lasla/opera-latina/`.

are the emotions conveyed by the poet in the sentence under analysis?

- `positive`: the only emotions that are conveyed at lexical level and the only images that are evoked are positive, or positive emotions are clearly prevalent;
- `negative`: the only emotions that are conveyed at lexical level and the only images that are evoked are negative, or negative emotions are clearly prevalent;
- `neutral`: there are no emotions conveyed by the text;
- `mixed`: lexicon and evoked images produce opposite emotions; it is not possible to find a clearly prevailing emotion.

The annotation of the GS was organized in four phases. In the first phase, annotators worked together collaboratively assigning the sentiment class to four of the eight odes (21 sentences): the task was discussed and a common procedure was defined. In the second phase, annotators worked independently on the other four odes (23 sentences): A1 and A2 annotated the original Latin text, while A3 annotated the same odes using an Italian translation (Horace and Nuzzo, 2009) to understand how the use of texts not in the original language can alter the annotation of the sentiment. In the third phase, we calculated the Inter-Annotator Agreement, whereas in the last phase disagreements were discussed and reconciled.

3.2 Inter-Annotator Agreement

Cohen's k between A1 and A2 resulted in 0.5, while Fleiss's k among the three annotators (A1-A2-A3) resulted in 0.48 (both these results are considered moderate agreement). In particular, the `negative` class proved to be the easiest to be annotated (with a Fleiss's k of 0.64), followed by `neutral` (0.57) and `positive` (0.45), whereas `mixed` was the most problematic class (0.23).

We noticed that the Italian translation was sometimes misleading, resulting in cases of disagreement: e.g., the sentence *inmortalia ne speres monet annus et almum quae rapit hora diem*, (ode IV, 7) is translated as 'speranze di eterno ti vietano gli anni e le ore che involano il giorno radioso' (literal translation of the Italian sentence into English: 'hopes of eternity forbid you the years and the hours that steal the radiant day'). A3 marked this sentence as `mixed`, considering that it is impossible to identify a prevailing emotion between the negativity expressed by the verb 'vietare' ('to forbid') and the positivity of 'giorno radioso' ('radiant day'). However, the translation of the Latin verb *rapio* is not appropriate: the Italian verb 'involare' ('to steal') does not convey the idea of the violent force inherent in *rapio*, which can be more correctly translated with the verb 'to plunder'.[7]

3.3 Reconciliation

Disagreements were discussed and reconciled by the three annotators: Table 1 presents the number of sentences and tokens per sentiment class. Our GS includes a majority of positive sentences (45.4%). Positive (average length: 21, standard deviation: 11), negative (average length: 24, standard deviation: 14), and mixed (average length: 25, standard deviation: 9) sentences are considerably longer than neutral ones (average length: 8, standard deviation: 3). Annotated examples are given in Table 2: English translations by Kaimowitz et al. (2008) are included for clarity.

	Sentences	**Tokens**
`positive`	20	411
`negative`	12	292
`neutral`	3	23
`mixed`	9	229
TOTAL	44	955

Table 1: Gold Standard statistics.

4 Experiments

4.1 Lexicon-Based Sentiment Analysis

The dataset for this experiment is obtained by means of a simple dictionary lookup of the lemmas in the *LatinAffectus* sentiment lexicon. Entries in the lexicon are assigned a score of: -1.0, -0.5 (negative polarity), 0 (neutral polarity), +0.5, +1.0 (positive polarity). The tokens in the *Odes* that are lemmatized under lemmas that also have an entry in the *LatinAffectus* are assigned the score that is found in the lexicon. For instance, the adjective *malus* 'bad' is found with a polarity value of -1.0 in *LatinAffectus*. All tokens lemmatized as *malus* (adj.) are thus given a score of -1.0. Note

[7]See for instance the English translation by Kaimowitz et al. (2008): "Do not hope for what's immortal, the year warns, and the hour which plunders the day".

Ode	Sent.	Text	Translation	Class
1.17	103	*hic tibi copia manabit ad plenum benigno ruris honorum opulenta cornu*	Here for you will flow abundance from the horn that spills the country's splendors	`positive`
4.7	549	*cuncta manus auidas fugient heredis amico quae dederis animo*	All that you bestow upon your heart escapes the greedy hands of an heir	`negative`
2.13	265	*frigora mitescunt Zephyris uer proterit aestas interitura simul pomifer autumnus fruges effuderit et mox bruma recurrit iners*	With the Zephyrs cold grows mild, summer tramples springtime, soon to die, once productive autumn pours forth its fruits, and shortly lifeless winter is back	`mixed`
2.7	235	*quem Venus arbitrum dicet bibendi*	Who will Venus name as master of the wine?	`neutral`

Table 2: Annotated examples taken from the Gold Standard.

that a score of 0.0 is assigned to both words expressly annotated as neutral in *LatinAffectus* and to those that do not have an entry in the lexicon.

The dictionary lookup required some manual disambiguation in cases of ambiguity due to homography. For 18 lemmas (corresponding to 49 tokens in the *Odes*), the sentiment lexicon provides multiple values; in most cases, as with *ales* 'winged' (adj.), but also 'bird' (n.), the variation is due to a different polarity attributed to the syntactic uses of the word (in the example, to the adjective and the noun). In such cases, the PoS annotation in the LASLA corpus was used to disambiguate and assign the correct score. We also reviewed those words that, although not tagged as nouns or adjectives in LASLA, still yield a match in *LatinAffectus*. After revision, we decided to keep the scores for a series of lemmas annotated as numerals in the corpus (*simplex* 'simple, plain', *primus* and *primum* 'first', *prius* 'former, prior') and the indefinite pronoun *solus* 'alone, only' that in *LatinAffectus* are marked as adjectives.

A sentence score (S) was computed by summing the values of all words. Thus, we attributed the label `positive` to all the sentences with score $S > 0$ and `negative` where $S < 0$. For $S = 0$, we attributed `neutral` to sentences where all words had a score of 0 and `mixed` where positive and negative words were equivalent. The overall accuracy of this method is 48% (macro-average F1 37, weighted macro-average F1 44) with unbalanced scores among the four classes: 70% for `positive`, 42% for `negative`, 67% `neutral`, while no correct predictions were given for `mixed`.

4.2 Zero-Shot Classification

We trained a language model for SA on English and tested it on our GS by relying on two state-of-the-art multilingual models. More specifically, we fine-tuned Multilingual BERT (mBERT) (Pires et al., 2019) and XLM-RoBERTa (Conneau et al., 2020) with the GoEmotions corpus (Demszky et al., 2020) using the Hugging Face's PyTorch implementation.[8] GoEmotions is a dataset of comments posted on Reddit manually annotated for 27 emotion categories or Neutral. In order to adapt this dataset to our needs, we mapped the emotions into sentiment categories as suggested by the authors themselves. For example, joy and love were converged into a unique `positive` class, whereas fear and grief were merged under the same `negative` class. The `neutral` category remained intact and comments annotated with emotions belonging to opposite sentiments were marked as `mixed`. Comments labeled with ambiguous emotions (i.e. realization, surprise, curiosity, confusion) were instead left out.[9] With this procedure, we built a training set made of 18,617 positive, 10,133 negative, 1,965 neutral and 1,581 mixed comments. For fine-tuning, we chose the

[8] `https://huggingface.co/transformers/index.html`

[9] For the full mapping, please see: `https://github.com/google-research/google-research/blob/master/goemotions/data/sentiment_mapping.json`.

Language	Test Set	Genre	mBERT	XLM-RoBERTa
English	GoEmotions	social media	86%	73%
	AIT-2018	social media	64%	59%
	Poem Sentiment	literary - poetry	50%	70%
Italian	MultiEmotions-It	social media	70%	75%
	AriEmozione	literary - opera	50%	52%
Latin	Horace GS	literary - poetry	32%	30%

Table 3: Accuracy of the mono-lingual and cross-lingual (zero-shot) classification method.

	Lexicon-Based SA			Zero-Shot mBERT			Zero-Shot XML-RoBERTa		
	P	R	F1	P	R	F1	P	R	F1
`positive`	0.56	0.70	0.62	0.83	0.25	0.38	1.00	0.10	0.18
`negative`	0.62	0.42	0.50	0.75	0.50	0.60	0.53	0.67	0.59
`neutral`	0.25	0.67	0.36	0.10	1.00	0.18	0.11	1.00	0.20
`mixed`	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 4: Precision (P), recall (R) and F1-score (F1) for the lexicon-based method and for the zero-shot classification experiment.

following hyperparameters: 32 for batch size, 2e-5 for learning rate, 6 epoches, AdamW optimizer.[10]

We evaluated the trained model on different datasets, including our GS. For each of the following test sets, we randomly selected 44 texts so to have the same number of input data as in our GS:

- GoEmotions: test set taken from the same corpus used for training the English model.
- Poem Sentiment: collection of English verses annotated with the same sentiment classes as in our GS (Sheng and Uthus, 2020).
- AIT-2018: English data of the emotion classification task of SemEval-2018 Task 1: Affect in Tweets (Mohammad et al., 2018). Each tweet is annotated as neutral or as one, or more, of eleven emotions. The original annotation was mapped onto our four sentiment classes, leaving out ambiguous emotions.
- AriEmozione: verses taken from 18th century Italian opera texts annotated with one or two emotions and the level confidence of the annotators (Fernicola et al., 2020). We randomly selected our test set from verses with high confidence scores, mapping emotions onto our four sentiment classes. Since the dataset does not contain verses annotated with opposite emotions, the class `mixed` is not present in the test set we built.
- MultiEmotions-It: a multi-labeled emotion dataset made of Italian comments posted on YouTube and Facebook (Sprugnoli, 2020). The original emotion labels were converted into our four classes.

Table 3 reports the results of mono-lingual and cross-lingual classification for the different datasets briefly described above and for the two pre-trained multilingual models. There is no clear prevalence of one model over the other: results vary greatly from one dataset to another. On the same language (thus without zero-shot transfer), we notice a drop in the performance for both mBERT and XML-RoBERTa when moving from Reddit comments, that is the same type of text as the training data, to tweets, but even more so when they are evaluated on poems. As for the zero-shot classification, results on Italian YouTube and Facebook comments are better than the ones registered on English tweets, but accuracy drops when applied to opera verses. However, the worst results are recorded for Latin with an accuracy equal to, or slightly above 30% (for mBERT: macro-average F1 29, weighted macro-average F1 35; for XML-RoBERTa: macro-average F1 24, weighted macro-average F1 26). For both mBERT and XML-RoBERTa, we register the same trend at class level: perfect accuracy for `neutral`, good accuracy for `negative` (50% with mBERT and 67% with XML-RoBERTa), low accuracy for `positive` (25% with mBERT and 10% with

[10]We adapted the following implementation: `https://gist.github.com/sayakmisra/b0cd67f406b4e4d5972f339eb20e64a5`.

XML-RoBERTa) and no correct predictions for mixed.

5 Conclusions and Future Work

In this paper we have presented a new GS, made of odes written by Horace, for the annotation of sentiment in Latin poetry. The extension of the manually annotated dataset is one of our future work: the goal is to have a sufficient amount of data to test supervised systems. We have also experimented two different SA approaches that do not require training data: both of them are not able to correctly identify sentences with mixed sentiments, which, in any case, are the most problematic also for human annotators. Table 4 reports a comparison in terms of precision, recall and F1-score among the lexicon-based approach and the zero-shot classification experiments with both the mBERT and the XML-RoBERTa models. The former performs better on the positive class whereas the zero-shot method achieves a higher F1-score on the negative one even if this class is not the most frequent in the training data. Both mBERT and XML-RoBERTa obtain a very high precision on the sentences marked as positive (0.83 and 1.00 respectively) but the recall is extremely low (0.25 and 0.10 respectively). On the contrary, for the neutral class, the recall is perfect (1.00 for both models) but the precision is very low (0.10 and 0.11 respectively).

A manual inspection of the output of the lexicon-based method revealed two main problems of that approach: i) the limited coverage of *LatinAffectus* and ii) sentiment shifters are not properly taken into consideration. As for the first point, *LatinAffectus* covers the 43% of nominal and adjectival lemmas in the GS, leaving out lemmas with a clear sentiment orientation. To overcome this issue, we are currently working on the extension of the lexicon with additional 10,000 lemmas. Regarding the sentiment shifters, their impact is exemplified by the following sentence: *cum semel occideris et de te splendida Minos fecerit arbitria non Torquate genus non te facundia non te restituet pietas* ('When you at last have died and Minos renders brillant judgement on your life, no Torquatus, not birth, not eloquence, not your devotion will bring you back.' - ode IV, 7). Here, the sentiment score calculated by the script is very positive (3) because it does not handle the frequent negations: however, the particle *non* should reverses the positive polarity of *facundia* 'eloquence' and *pietas* 'devotion'. This problem could be mitigated by modifying the script with rules that take into account negations and their focus.

Regarding the zero-shot classification approach, the very low performances on Latin deserve further investigation. It is possible that the problem lies in the data used to build the pre-trained models: i.e., Wikipedia for mBERT and Commoncrawl for XML-RoBERTa. Both resources were developed by relying on automatic language detection engines and are highly noisy due to the presence of languages other than Latin and of terms related to modern times. An additional improvement may also come from using for fine-tuning an annotated in-domain corpus in a well-resource language, that is a corpus of annotated poems: unfortunately, the currently available corpora are not big enough for such purpose.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme – Grant Agreement No. 769994.

References

Ouais Alsharif, Deema Alshamaa, and Nada Ghneim. 2013. Emotion classification in arabic poetry using machine learning. *International Journal of Computer Applications*, 65(16).

Gianluigi Baldo. 2012. Horace (Quintus Horatius Flaccus), Carmina. In Christine Walde and Brigitte Egger, editors, *Brill's New Pauly Supplements I - Volume 5 : The Reception of Classical Literature*. Brill, Amsterdam, October. Publisher: Brill.

Linda Barros, Pilar Rodriguez, and Alvaro Ortigosa. 2013. Automatic Classification of Literature Pieces by Emotion Detection: A Study on Quevedo's Poetry. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*, pages 141–146. IEEE.

Venkateswarlu Bonta and Nandhini Kumaresh2and N Janardhan. 2019. A comprehensive study on lexicon based approaches for sentiment analysis. *Asian Journal of Computer Science and Technology*, 8(S2):1–6.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July. Association for Computational Linguistics.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A Dataset of Fine-Grained Emotions. In *58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Francesco Fernicola, Shibingfeng Zhang, Federico Garcea, Paolo Bonora, and Alberto Barrón-Cedeño. 2020. AriEmozione: Identifying Emotions in Opera Verses. In *Proceedings of the Seventh Italian Conference on Computational Linguistics (CLiC-it 2020)*. Accademia University Press.

Thomas Haider, Steffen Eger, Evgeny Kim, Roman Klinger, and Winfried Menninghaus. 2020. PO-EMO: Conceptualization, annotation, and modeling of aesthetic emotions in German and English poetry. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1652–1663, Marseille, France, May. European Language Resources Association.

Horace and Gianfranco Nuzzo. 2009. *I quattro libri delle Odi e l'Inno secolare di Quinto Orazio Flacco*. Flaccovio.

Yufang Hou and Anette Frank. 2015. Analyzing sentiment in classical chinese poetry. In *Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH)*, pages 15–24.

Philip N. Johnson-Laird and Keith Oatley. 2016. Emotions in music, literature, and film. In Lisa Feldman Barrett, Michael Lewis, and Jeannette M. Haviland-Jones, editors, *Handbook of emotions*, chapter 3, pages 82–97. The Guildford Press.

Jeffrey H Kaimowitz, Ronnie Ancona, et al. 2008. *The odes of Horace*. Johns Hopkins University Press.

Evgeny Kim and Roman Klinger. 2018. A survey on sentiment and emotion analysis for computational literary studies. *arXiv preprint arXiv:1808.03137*.

Walaa Medhat, Ahmed Hassan, and Hoda Korashy. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal*, 5(4):1093–1113.

Saif M. Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 Task 1: Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA.

Gaurav Mohanty, Pruthwik Mishra, and Radhika Mamidi. 2018. Kabithaa: An annotated corpus of odia poems with sentiment polarity information. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Paris, France, may. European Language Resources Association (ELRA)*.

Marco Passarotti, Francesco Mambrini, Greta Franzini, Flavio Massimiliano Cecchini, Eleonora Litta, Giovanni Moretti, Paolo Ruffolo, and Rachele Sprugnoli. 2020. Interlinking through lemmas. the lexical collection of the LiLa knowledge base of linguistic resources for Latin. *Studi e Saggi Linguistici*, 58(1):177–212.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How Multilingual is Multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001.

Emily Sheng and David C Uthus. 2020. Investigating societal biases in a poetry composition system. In *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, pages 93–106.

Rachele Sprugnoli and Marco Passarotti, editors. 2020. *Proceedings of LT4HALA 2020 - 1st Workshop on Language Technologies for Historical and Ancient Languages*, Marseille, France, May. European Language Resources Association (ELRA).

Rachele Sprugnoli, Francesco Mambrini, Giovanni Moretti, and Marco Passarotti. 2020a. Towards the Modeling of Polarity in a Latin Knowledge Base. In *Proceedings of the Third Workshop on Humanities in the Semantic Web (WHiSe 2020)*, pages 59–70.

Rachele Sprugnoli, Marco Passarotti, Daniela Corbetta, and Andrea Peverelli. 2020b. Odi et Amo. Creating, Evaluating and Extending Sentiment Lexicons for Latin. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3078–3086.

Rachele Sprugnoli. 2020. MultiEmotions-it: A new dataset for opinion polarity and emotion analysis for Italian. In *Proceedings of the Seventh Italian Conference on Computational Linguistics (CLiC-it 2020)*, pages 402–408. Accademia University Press.

PS Sreeja and GS Mahalakshmi. 2019. Perc-an emotion recognition corpus for cognitive poems. In *2019 International Conference on Communication and Signal Processing (ICCSP)*, pages 0200–0207. IEEE.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253.

Playing with NeMo for Building an Automatic Speech Recogniser for Italian

Fabio Tamburini
FICLIT - University of Bologna, Italy
fabio.tamburini@unibo.it

Abstract

This paper presents work in progress for the creation of a Large Vocabulary Automatic Speech Recogniser for Italian using NVIDIA NeMo. Thanks to this package, we were able to build a reliable recogniser for adults' speech by fine tuning the English model provided by NVIDIA and rescoring it with powerful neural language models, obtaining very good performances. The lack of a standard, reliable and publicy available baseline for Italian motivated this work.

1 Introduction

The advent of the "Deep Learning Revolution" introduced astonishing changes also in the field of speech processing allowing for the development of brand new tools and devices able to recognise and synthesise speech exhibiting performances never seen before. It is sufficient to think to the new virtual assistants that populates our houses and mobile phones for getting an immediate idea about the improvements in this research field.

Most big IT companies developed, in the past 3/4 years, solutions well integrated with various devices that include high performance tools for speech processing. However, these solutions very often are not released freely, sometimes they require registrations and fees and, in the best situations, codes are free, but the models for a specific language are not available. A notable exception regards NVIDIA NeMo[1], a conversational AI toolkit built for researchers working on Automatic Speech Recognition (ASR), Natural Language Processing (NLP), and Text-To-Speech synthesis (TTS). The primary objective of NeMo is to help researchers from industry and academia to reuse prior work, namely code and pretrained models for various languages, and make it easier to create new conversational AI models, maybe adapting tools and models to specific languages or particular domains.

This paper reports an attempt to build a high performance Large Vocabulary ASR system for Italian adults' speech by exploiting all the features available in NeMo and most of the largest Italian spoken corpora available to the community.

Section 2 describes the various speech datasets used for developing the model, followed by Section 3 that describes the state of the art; in Section 4 we will describe the NeMo ASR model used in the experiments and Section 5 will discuss the experiments and the obtained results. Section 6 draws some provisional conclusions about our work.

2 Italian Spoken Corpora for ASR

This section describes the datasets we used for the creation of the Italian ASR model. We have to say that, of course, these are not the only spoken corpora available, but they are the biggest corpora commonly used for setting up an ASR system for Italian. They are typically very big, already organised and structured exactly for training ASR systems or specifically designed to maximise their impact and usefulness for ASR. We have also to say that, as far as we know, this is the first attempt to use all of them for ASR training in a single project.

2.1 Mozilla Common Voice (v7.0)

Common Voice (Ardila et al., 2020) is a crowdsourcing project started by Mozilla to create a free database for setting up speech recognition software. The project is supported by volunteers who record sample sentences with a microphone and review recordings of other users. The transcribed

[1]Other exceptions providing also multilingual models including Italian are Facebook Wav2Vec and SpeechBrain.

utterances will be collected in a voice database available under the public domain license CC0. This license ensures that developers can use the database for voice-to-text applications without restrictions or costs.

With regard to the Italian subcorpus, they currently[2] released version 7.0 (MCV7), containing 6,407 speakers for a total of 160,570 utterances with the correct transcriptions. In the standard splitting provided with the dataset the training set contains 131,041 utterances corresponding to 189.50 hours of speech, the validation set 14,764 utterances for 24.41 hours and the test set 14,765 utterances corresponding to 25.74 hours.

These splitting are very important for our experiments, as discussed in Section 5.

2.2 Multilingual LibriSpeech

Multilingual LibriSpeech[3] (MLS) dataset (Pratap et al., 2020) is a large multilingual corpus suitable for speech research. The dataset is derived from read audiobooks from LibriVox and consists of 8 languages - English, German, Dutch, Spanish, French, Italian, Portuguese and Polish. The Italian section contains 42,935 utterances for a total of 160.06 hours of transcribed speech.

2.3 VoxForge

VoxForge[4] is an open speech dataset that was set up to collect transcribed speech for use with Free and Open Source Speech Recognition Engines. The Italian portion of VoxForge contains 10,633 utterances totalling 20.16 hours of transcribed speech.

2.4 APASCI

APASCI (Angelini et al., 1994) is an Italian speech database recorded in an insulated room with a Sennheiser MKH 416 T microphone. The speech material, consisting of 2,170 utterances with a wide phonetic/diphonic coverage and totalling 2.91 hours of speech, was read by 100 Italian speakers (50 male and 50 female). The database includes the transcription of each utterance both at phonemic and at orthographic levels. This database in the past allowed to design, train and evaluate continuous speech recognition systems (speaker independent, speaker adaptive, speaker dependent, multispeakers). It was also designed for research on acoustic modelling as well as on acoustic parameters for speech recognition and for research on speaker recognition.

[2]July 2021.

[3]`http://www.openslr.org/94/`

[4]`http://www.voxforge.org/`

3 State of the Art for Italian ASR

In order to properly describe the state of the art, we should first define the typical metrics used for evaluating ASR systems. Given the system transcription for an utterance and the correct transcription extracted from the gold standard, the most important metric is certainly the Word Error Rate (WER) defined as

$$WER = \frac{(Insertions + Substitutions + Deletions)}{Gold\ Number\ of\ Words},$$

typically expressed in percentage. It compares the two transcriptions counting all the differences at word level using the edit distance between them. We can also define the Phone Error Rate (PER) and the Character Error Rate (CER) that use the same principle but applied, respectively, at phone or character level.

Examining the literature for the construction of ASR models for Italian we immediately recognise a lack of works devoted to the building of a general Large Vocabulary ASR for adults' speech. The only work we found on that was presented by Cosi and Hosom (2000), used a rather old approach to the problem (a hybrid HMM/ANN architecture) and measures the performance only on phones and not on words. Using PER instead the most common WER is a common trait of all the subsequent works we found in literature (Cosi and Pellom, 2005; Cosi, 2008; Cosi et al., 2014; Cosi, 2015) that applied a lot of different system architectures only on child speech. This large bundle of works represent the main line of research for building Italian ASR systems, but the aim of these studies is completely different from ours and, moreover, their results are not directly comparable with ours.

An exception to what we said before is represented by the work of Gretter (2014): he first built a large multilingual benchmark corpus, extracting data from the portal Euronews, consisting of about 100 hours of adults' speech for each language and, second, he developed also some ASR baselines, based on triphone Hidden Markov Models and n-gram Language models, obtaining on Italian a word recognition accuracy of 83.5% leading to a

WER=16.5%, a quite remarkable result obtained using non-neural stochastic systems.

More recent studies employing neural models were able to build other quite reliable systems. Weibin (2019) trained a system based on DeepSpeech (Hannun et al., 2014) using VoxForge, CLIPS[5], SI-CALLIOPE (Tedesco et al., 2018), LibriVox Audiobooks[6] and Mozilla Common Voice corpora for a total of 438 hours of speech, obtaining a WER=13.8% on a mixed test set. Pratap et al. (2020) made some experiments using wav2letter++[7] followed by a 5gram rescoring obtaining a test WER=28.19%. They used different test sets w.r.t. the one used in this work, thus they can only provide some general indications about WER, but they are not directly comparable to our work.

4 NVIDIA NeMo ASR

Traditional speech recognition takes a generative approach, modelling the recognition process of speech sounds acoustics (O) as $\overline{W} = \mathrm{argmax}_W\, P(O|W)P(W)$ where W is a possible transcription as sequence of words. The actors of the game include a language model $P(W)$ that allows to estimate the most likely orderings of words in a given language (e.g. an n-gram model), a pronunciation model for each word in that sequence (e.g. a lexicon of phonetically transcribed words) and an acoustic model $P(O|W)$ that allows to estimate the probability of an input sequence of acoustic observations given each possible words sequence W. When we receive some spoken input, our goal would be to find the most likely sequence of text that maximises the words probability given a speech-acoustic input.

Over time, neural nets advanced to the point where each component of the traditional speech recognition model could be replaced by a neural model that had better performance and that had a greater potential for generalisation. For example, we could replace an n-gram model with a neural language model, and replace a pronunciation table with a neural pronunciation model, and so on. However, each of these neural models need to be trained individually on different tasks, and errors in any model in the pipeline could throw off the whole prediction.

Nowadays, end-to-end ASR discriminative architectures models that simply take a sequence of audio inputs and give a sequence of textual outputs, and in which all components of the architecture are trained jointly towards the same goal, largely dominate the field. The model's encoder would be akin to an acoustic model for extracting speech features, which can then be directly piped to a decoder which directly outputs text, as a sequence of characters, in a given language. If desired, we could still integrate a language model that would improve our predictions, piping it after the decoder[8].

Grasping information from NeMo github site[9], we learn that the base ASR model provided by NVIDIA is *Jasper* ("Just Another Speech Recognizer") (Li et al., 2019) a deep Time Delay Neural Network comprising of blocks of 1D-convolutional layers. The Jasper family of models are denoted as "Jasper_[BxR]" where B is the number of blocks and R is the number of convolutional sub-blocks within a block. Each sub-block contains a 1-D convolution, batch normalisation, ReLU, and dropout.

Most state-of-the-art ASR models are extremely large; they tend to have on the order of a few hundred million parameters. This makes them hard to deploy on a large scale given current limitations of devices on the edge. Another model is included into NeMo, *QuartzNet* (Kriman et al., 2020), a version of Jasper with separable convolutions and larger filters. It can achieve performance similar to Jasper but with an order of magnitude fewer parameters. Similarly to Jasper, the QuartzNet family of models are denoted as "QuartzNet_[BxR]", where B is the number of blocks and R is the number of convolutional sub-blocks within a block, and do not use the computationally costly recurrent layers in favour of more efficient convolutional layers. Each sub-block contains a 1-D separable convolution, batch normalisation, ReLU, and dropout (see Figure 1 for a complete diagram describing the QuartzNet internal structure). Both models described before optimise the Connectionist Temporal Classification (CTC) loss.

NVIDIA provided also a large number of pre-

[5] http://www.clips.unina.it

[6] https://librivox.org/

[7] https://github.com/flashlight/wav2letter

[8] Partially taken from, https://docs.nvidia.com/deeplearning/nemo/user-guide/docs/en/main/asr/intro.html

[9] https://github.com/NVIDIA/NeMo

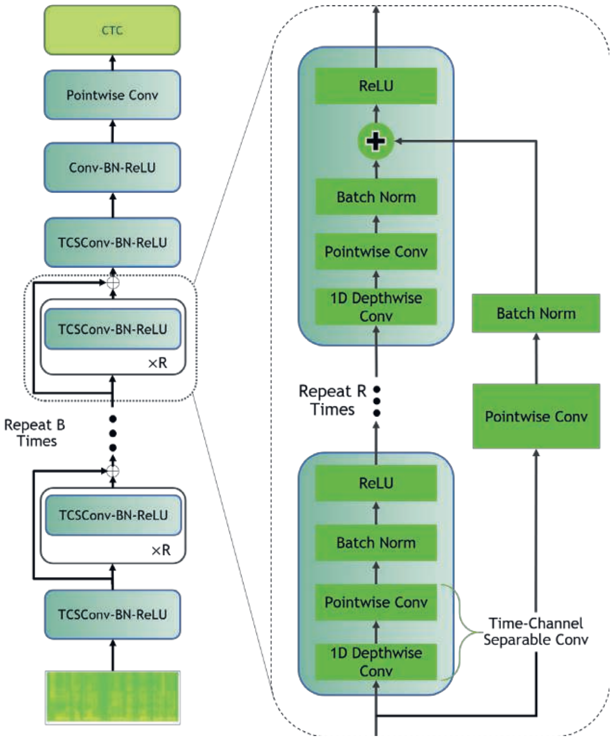


Figure 1: NVIDIA NeMo QuartzNet model.

trained models[10] for various languages. The two models for English "STT_en_Quartznet15x5" and Italian "STT_it_Quartznet15x5" (both at version 1.0.0rc1 published the 30th June 2021) are relevant for our work. The Quartznet 15x5 model family consists of 79 layers and has a total of 18.9 million parameters, with five blocks that repeat fifteen times plus four additional convolutional layers.

QuartzNet15x5 Encoder and Decoder English neural module's checkpoints from NVIDIA were trained using Multilingual LibriSpeech and Mozilla's English Common Voice 6.1 "validated" set (a huge amount of data containing more than 3,300 hours of speech) with two types of data augmentation techniques: speed perturbation and Cutout. Speed perturbation means that additional training samples were created by slowing down or speeding up the original audio data by 10%. Cutout refers to randomly masking out small rectangles out of the spectrogram input as a regularization technique. NVIDIA's Apex/Amp O1 optimization level was used for training achieveing 4.19% WER on LibriSpeech test-clean.

NeMo documentation also describes a procedure for fine-tuning the English model to adapt it to other languages, keeping the acoustic encoder frozen and fine-tuning the decoder for producing transcriptions for a different language (Huang et al., 2020). In the cited paper they also get the relevant conclusion that it is much better, in terms of performance, to fine-tune the English model than to retrain from scratch a new model for a specific language. The Italian model provided by NVIDIA has been produced following the suggested procedure, in particular by retraining the QuartzNet decoder using the training portion of MCV version 6.1. We will consider this Italian model as a baseline for our experiments.

5 Model Setup and Results

The STT_it QuartzNet model provided by NVIDIA was trained using a reduced set of data and applying an output dictionary that includes some characters that do not belong to the Italian alphabet. For these reasons we preferred to restart the fine-tuning process directly from the original STT_en_Quartznet15x5 English model.

The training set we used to fine tune the NVIDIA STT_en model to Italian is composed by joining the training portion of MCV7 and all files from MLS, VoxForge and APASCI, and contains 186,778 utterances/speech files totalling 372.62 hours of transcribed speech. 19,199 utterance/files were filtered out from the training set totalling 97.77 hours of removed speech. This is due to the fact that in some dataset, mainly in MLS and VoxForge, there were some utterances longer than 16.7 seconds, a time limit hard coded into NeMo in order to keep the model computationally tractable. We checked also that transcriptions contain only the 34 standard characters from the Italian alphabet (26 lowercase letters plus six accented characters, the apostrophe and the space) as it is a standard practice in ASR to lowercase transcriptions and to remove any punctuation mark not strictly useful or relevant to help the recognition.

With regard to decoding and rescoring, NeMo offers various possibilities:

- **Greedy Decoding**. This method simply computes the most likely sequence of characters, also called as the "best-path decoder", given the audio input.

- **Beam Search Decoding**. Beam Search Decoding (BSD) is another way of decoding model prediction that leads to better results than the greedy search. BSD, instead of choosing al-

[10] `https://ngc.nvidia.com/catalog/collections/nvidia:nemo_asr`

ways the best prediction at each step, considers the top-K hypothesis having the highest probabilities, where K is the so called *beam size*. For all the subsequent experiments we used beam_size=1024, beam_alpha=1.0 and beam_beta=0.5 (see NeMo documentation).

Language Models (LM) have shown to help the accuracy of ASR models when combined to BSD. NeMo currently supports the following two approaches to incorporate language models into the ASR models through BSD:

– **N-gram Rescoring**. In this approach, an N-gram Language Model is trained on text data, then it is used in fusion with beam search decoding to find the best candidates. The beam search decoders in NeMo support language models trained with the KenLM library (Heafield et al., 2013). We used this library code for building a 3-gram and a 6-gram LM using the 165-million-token-version of the CORIS corpus[11] (Rossini Favretti et al., 2002) specially cleaned and prepared for this task.

– **Neural Rescoring**. In the neural rescoring approach a neural network is used to give scores to a candidate text transcript predicted by the decoder of the ASR model. The top K candidates produced by the beam search decoding are given to a neural language model to rank them. This score is usually combined with the scores from the beam search decoding to produce the final scores and rankings. NeMo neural LMs are based on the Transformer sequence-to-sequence architecture like those described in (Vaswani et al., 2017). Again, we used the CORIS corpus described above to train an Italian neural LM from scratch and, after a month of training, we reached a perplexity of 29.30.

Given such possibilities, we fine tuned the STT_en model on a single V100 GPU using our joined dataset described above and the MCV7 validation and test set respectively for early stopping the training process and to evaluate all models. The hyperparameters we modified w.r.t. the original English model, and contained in the model itself, are listed in Table 1.

As notable exception to the NVIDIA suggested procedure for fine tuning a model, we have to report that we obtained the best results by unfreezing the encoder and letting it to slightly adapt the extracted speech features to the new language, namely Italian, that certainly share most of the sounds with the starting English model STT_en, but contains also specific sounds (e.g. [ɲ] and [ʎ]) that may require small adaptations.

[11] Corresponding to the 2021 brand new update.

Par.	**Value**
train_ds.batch_size	96
validation_ds.batch_size	4
optim.lr	0.0012
optim.betas	[0.8,0.5]
optim.weight_decay	0.001
optim.warmup_steps	500
optim.sched.min_lr	1e-10
trainer.precision	16
trainer.amp_level	O1

Table 1: Hyperparametrs modified during the fine-tuning process w.r.t. the STT_en_Quartznet15x5 model.

Table 2 outlines our results after a complete fine tuning of the end-to-end ASR model using the Italian dataset described before and applying different decoding and rescoring schemas. The improvement obtained with the fine-tuning process, when compared to the original model delivered by NVIDIA is relevant, but not so big, while when applying the BSD with the two rescoring algorithms the WER metric improve of 40% w.r.t. the greedy decoding schema.

System	**Valid.**	**Test**
Baseline (NVIDIA STT_it)		
Greedy Decoding	15.64/4.00	16.90/4.46
BSD & 3-gram Resc.	10.79/3.18	11.59/3.54
BSD & 6-gram Resc.	10.77/3.17	11.57/3.53
BSD & Neural Resc.	9.54/ -	10.51/ -
NVIDIA STT_en + Our Retraining		
Greedy Decoding	14.86/3.78	15.82/4.14
BSD & 3-gram Resc.	10.41/2.97	10.96/3.27
BSD & 6-gram Resc.	10.36/2.95	10.94/3.26
BSD & Neural Resc.	**9.04/ -**	**9.67/ -**

Table 2: WER/CER results (in percentage) on Mozilla Common Voice v7.0 (MCV7) validation and test sets.

6 Conclusions

This paper presented work in progress for the construction of a reliable and performing ASR system for Italian adults' speech. Thanks to the NVIDIA NeMo package, we were able to produce a very strong baseline reaching a WER = 9.67% over the MCV7 test set.

This is only the beginning of our work, as any change in the kind of speech used to train the system could degrade the whole performance, but, having used a collection of four different datasets containing thousands of different speakers and speech utterances for setting up such ASR system, we believe that the result should be robust enough. Unfortunately, the lack of a standardised benchmark for Italian does not allow for a quantitative and objective evaluation of this statement.

End-to-end character ASR model, and its improvement on WER, is only part of the game: the work on decoding and rescoring procedures produced much more improvements. Thus, the most important "take home lesson" is certainly to focus on the development of high performance LM specifically tuned for ASR.

All the models presented in this paper as well as the scripts and additional codes for using NeMo and generating the results will be made available[12].

Acknowledgements

We acknowledge the CINECA[13] award no. HP10C7XVUO (project QT4CLML) under the ISCRA initiative, for the availability of HPC resources and support.

References

B. Angelini, F. Brugnara, D. Falavigna, D. Giuliani, R. Gretter, and M. Omologo. 1994. Speaker Independent Continuous Speech Recognition Using An Acoustic-Phonetic Italian Corpus. In *Proc. of the 3rd International Conference on Spoken Language Processing - ICSLP '94*, pages 1391–1394, Yokohama, Japan.

R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proc. of the 12th Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France. European Language Resources Association.

P. Cosi. 2008. Recent Advances in Sonic Italian Children's Speech Recognition for Interactive Literacy Tutors. In *Proc. 1st Workshop on Child, Computer and Interaction (WOCCI '08)*, Chania, Crete, Greece.

P. Cosi. 2015. A kaldi-dnn-based asr system for italian. In *Proc. 2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–5.

P. Cosi and J.P. Hosom. 2000. High performance "general purpose" phonetic recognition for italian. In *Sixth International Conference on Spoken Language Processing, ICSLP 2000/Interspeech 2000, Beijing, China, October 16-20, 2000*, pages 527–530. ISCA.

P. Cosi, M. Nicolao, G. Paci, G. Sommavilla, and T. Tesser. 2014. Comparing open source ASR toolkits on Italian children speech. In *Proc. 4th Workshop on Child Computer Interaction (WOCCI 2014)*.

P. Cosi and B.L. Pellom. 2005. Italian children's speech recognition for advanced interactive literacy tutors. In *Proc. Interspeech 2005*, pages 2201–2204.

R. Gretter. 2014. Euronews: a multilingual benchmark for ASR and LID. In *Proc. of the 15th Annual Conference of the International Speech Communication Association - INTERSPEECH 2014*, pages 1603–1607, Singapore. ISCA.

A.Y. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A.Y. Ng. 2014. Deep speech: Scaling up end-to-end speech recognition. *CoRR*, abs/1412.5567.

K. Heafield, I. Pouzyrevsky, J.H. Clark, and P. Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proc. of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 690–696, Sofia, Bulgaria. Association for Computational Linguistics.

J. Huang, O. Kuchaiev, P. O'Neill, V. Lavrukhin, J. Li, A. Flores, G. Kucsko, and B. Ginsburg.

[12]https://github.com/ftamburin/ItaNeMoASR

[13]https://www.cineca.it/en

2020. Cross-language transfer learning, continuous learning, and domain adaptation for end-to-end automatic speech recognition. *CoRR*, abs/2005.04290.

S. Kriman, S. Beliaev, B. Ginsburg, J. Huang, O. Kuchaiev, V. Lavrukhin, R. Leary, J. Li, and Y. Zhang. 2020. Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2020)*, pages 6124–6128.

J. Li, V. Lavrukhin, B. Ginsburg, R. Leary, O. Kuchaiev, J.M. M. Cohen, H. Nguyen, and R.T. Gadde. 2019. Jasper: An End-to-End Convolutional Neural Acoustic Model. In *Proc. Interspeech 2019*, pages 71–75.

V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert. 2020. MLS: A large-scale multilingual dataset for speech research. In *Proc. of , 21st Annual Conference of the International Speech Communication Association (Interspeech 2020)*, pages 2757–2761, Shanghai, China.

R. Rossini Favretti, F. Tamburini, and C. De Santis. 2002. CORIS/CODIS: A corpus of written Italian based on a defined and a dynamic model. In A. Wilson, P. Rayson, and T. McEnery, editors, *A Rainbow of Corpora: Corpus Linguistics and the Languages of the World*, pages 27–38. Lincom-Europa, Munich.

R. Tedesco, S. Cenceschi, and L. Sbattella. 2018. Verso il riconoscimento automatico della prosodia. In *Proc. AISV 2018*, pages 433–439.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L Kaiser, and I. Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems - NIPS'17*, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

C. Weibin. 2019. *Phoenix: Deep Speech Based Automatic Speech Recognition System for Italian Language*. Master Thesis, Politecnico di Milano.

OCTIS 2.0: Optimizing and Comparing Topic Models in Italian Is Even Simpler!

Silvia Terragni and Elisabetta Fersini
University of Milano-Bicocca, Milan, Italy
s.terragni4@campus.unimib.it, elisabetta.fersini@unimib.it

Abstract

English. OCTIS is an open-source framework for training, evaluating and comparing Topic Models. This tool uses single-objective Bayesian Optimization (BO) to optimize the hyper-parameters of the models and thus guarantee a fairer comparison. Yet, a single-objective approach disregards that a user may want to simultaneously optimize multiple objectives. We therefore propose OCTIS 2.0: the extension of OCTIS that addresses the problem of estimating the optimal hyper-parameter configurations for a topic model using multi-objective BO. Moreover, we also release and integrate two pre-processed Italian datasets, which can be easily used as benchmarks for the Italian language.

Italiano. *OCTIS è un framework open-source per il training, la valutazione e la comparazione di Topic Models. Questo strumento utilizza l'ottimizzazione Bayesiana (BO) a singolo obiettivo per ottimizzare gli iperparametri dei modelli e quindi garantire una comparazione più equa. Tuttavia, questo approccio ignora che un utente potrebbe voler ottimizzare pi'u di un obiettivo. Proponiamo perciò OCTIS 2.0: l'estensione di OCTIS che affronta il problema della stima delle configurazioni ottimali degli iperparametri di un topic model usando la BO multi-obiettivo. In aggiunta, rilasciamo e integriamo anche due nuovi dataset in italiano pre-processati, che possono essere facilmente utilizzati come benchmark per la lingua italiana.*

1 Introduction

Topic models are statistical methods that aim to extract the hidden topics underlying a collection of documents (Blei et al., 2003; Blei, 2012; Boyd-Graber et al., 2017). Topics are often represented by sets of words that make sense together, e.g. the words "cat, animal, dog, mouse" may represent a topic about animals. Topic models' evaluations are usually limited to the comparison of models whose hyper-parameters are held fixed (Doan and Hoang, 2021; Terragni et al., 2020a; Terragni et al., 2020b). However, hyper-parameters can have an impressive impact on the models' performance and therefore fixing the hyper-parameters prevents the researchers from discovering the best topic model on the selected dataset.

Recently, OCTIS (Terragni et al., 2021a, Optimizing and Comparing Topic Models is Simple) has been released: a comprehensive and open-source framework for training, analyzing, and comparing topic models, over several datasets and evaluation metrics. OCTIS determines the optimal hyper-parameter configuration according to a Bayesian Optimization (BO) strategy (Archetti and Candelieri, 2019; Snoek et al., 2012; Galuzzi et al., 2020). The framework already provides several features and resources, among which at least 8 topic models, 4 categories of evaluation metrics, and 4 pre-processed datasets. However, the framework uses a single-objective Bayesian optimization approach, disregarding that a user may want to simultaneously optimize more than one objective (Terragni and Fersini, 2021). For example, a user may be interested in obtaining topics that are coherent but also diverse and separated from each other.

Contributions. In this paper, we propose OCTIS 2.0, an extension of the existing framework that integrates both a single-objective and multi-objective hyper-parameter optimization strategy, using Bayesian optimization. Moreover, we also pre-process and include two novel datasets in Italian. We will then briefly show the potentiality of the extended framework by comparing different topic models on the new released Italian datasets. We believe these resources can be useful for the topic modeling and NLP communities, since they can be used as benchmarks for the Italian language.

2 OCTIS: Optimizing and Comparing Topic Models Is Simple!

2.1 OCTIS 1.0

OCTIS (Terragni et al., 2021a, Optimizing and Comparing is Simple!) is an open-source evaluation framework for the comparison of topic models, that allows a user to optimize the models' hyper-parameters for a fair experimental comparison. The evaluation framework is composed of different modules that interact with each other: (1) dataset and pre-processing tools, (2) topic modeling, (3) hyper-parameter optimization, (4) evaluation metrics. OCTIS can be used both as a python library and through a web dashboard. It also provides a set of pre-processed datasets, state-of-the-art topic models and several evaluation metrics.

We will now briefly describe the two components that we will extend in this work: the pre-processed datasets and the hyper-parameter optimization module.

Pre-processing and Datasets. OCTIS currently provides functionalities for pre-processing the texts, which include the lemmatization of the text, the removal of punctuation, numbers and stop-words, and the removal of words based on their frequency. Moreover, the framework already provides 4 pre-processed datasets, that are ready to use for topic modeling. These datasets are 20 NewsGroups,[1] M10 (Lim and Buntine, 2014), DBLP,[2] and BBC News (Greene and Cunningham, 2006). All the datasets are split into three partitions: training, testing and validation.

All the currently provided datasets are in English. OCTIS already provides language-specific pre-processing tools (e.g. lemmatizers for multiple languages), but it does not present datasets in other languages. Creating benchmark datasets for other languages is useful for investigating the peculiarities of different topic modeling methods.

Single-Objective Hyper-parameter Optimization. OCTIS uses single-objective Bayesian Optimization (Snoek et al., 2012; Shahriari et al., 2015) to tune the topic models' hyper-parameters with respect to a selected evaluation metric. In particular, the user specifies the search space for the hyper-parameters and an objective metric. Then, BO sequentially explores the search space to determine the optimal hyper-parameter configuration. Since the models are usually probabilistic and can give different results with the same hyper-parameter configuration, the objective function is computed as the median of a given number of model runs (i.e., topic models run with the same hyper-parameter configuration) computed for the selected evaluation metric. OCTIS uses the Scikit-Optimize library (Head et al., 2018) for the implementation of the single-objective hyper-parameter Bayesian optimization.

The use of a single-objective approach is however limited. In fact, this strategy disregards other objectives. For example, a user may require to optimize the coherence of the topics and their diversity at the same time.

2.2 OCTIS 2.0

New dataset resources for the Italian language. Since OCTIS provides only English datasets, we extend the set of datasets by including two new datasets in Italian. We build the two datasets from the Italian version of the Europarl dataset[3] and from the Italian abstracts of DBPedia.[4] In particular, we randomly sample 5000 documents from Europarl and we randomly sample 1000 Italian abstracts for 5 DBpedia types (event, organization, place, person, work), for a total of 5000 abstracts.

We preprocess the datasets using the following strategy: we lemmatize the text, we remove the punctuation, numbers and Italian stop-words, we filter out the words with a document frequency higher than the 50% and less than the 0.1% for Europarl and 0.2% for DBPedia and we also remove the documents with less than 5 words. These values have been chosen by manually inspecting the resulting pre-processed datasets.

We report the most relevant statistics of the

[1] http://people.csail.mit.edu/jrennie/20Newsgroups/

[2] https://github.com/shiruipan/TriDNR/tree/master/data

[3] https://www.statmt.org/europarl/

[4] https://www.dbpedia.org/resources/ontology/

novel Italian datasets in Table 1. Following the original paper, we split the datasets in three partitions: training (75%), validation (15%), and testing (15%).

Dataset	Num. of documents	Avg. doc length (Std. dev.)	Num. of unique words
DBPedia	4251	5.5 (11.8)	2047
Europarl	3616	20.6 (19.3)	2000

Table 1: Statistics of the pre-processed datasets.

From Single-objective to Multi-objective Hyper-parameter Bayesian Optimization. Given the limitations of the single-objective hyperparameter optimization approach, we extend OCTIS by including a multi-objective approach (Kandasamy et al., 2020; Paria et al., 2019). Single-objective BO can be in fact generalized to multiple objective functions, where the final aim is to recover the Pareto frontier of the objective functions, i.e. the set of Pareto optimal points. A point is Pareto optimal if it cannot be improved in any of the objectives without degrading some other objective. Using a multi-objective hyper-parameter optimization approach thus allows us not only to identify the best performing model, but also to empirically discover competing objectives.

Since the original Scikit-Optimize library does not provide multi-objective optimization tools, we use the *dragonfly* library[5] (Paria et al., 2019). Like the single-objective optimization, the user must specify the hyper-parameter search space. But in addition, they also need to specify which functions they want to optimize. We report a simple coding example below:

```
# loading of a pre-processed dataset
dataset = Dataset()
dataset.fetch_dataset("DBPedia_IT")

#model instantiation
lda = LDA(num_topics=25)

#definition of the metrics to optimize
td = TopicDiversity()
coh = Coherence()
metrics = [td, coh]

#definition of the search space
config_file = "path/to/search/space/file"
```

[5]https://github.com/dragonfly/dragonfly

```
#define and launch optimization
mmm = MOOptimizer(
        dataset=dataset, model=model,
        config_file=config_file,
        metrics=metrics, maximize=True)
mmm.optimize()
```

The snippet will run a multi-objective optimization experiment that will return the Pareto front of the diversity and coherence metrics on the Italian dataset DBPedia by optimizing the hyper-parameters (defined in a configuration file) of LDA with 25 topics.

In keeping with the spirit of the first version of OCTIS, the framework extension is open-source and easily accessible, in order to guarantee researchers and practitioners a fairer, accessible and reproducible comparison between the models (Bianchi and Hovy, 2021). OCTIS 2.0 is available as extension of the original library, at the following link: `https://github.com/mind-Lab/octis`.

3 Experimental Setting

In the following, we will show the capabilities of the extended framework on the new datasets by carrying out a simple experimental campaign.

We assume an experimental setting in which a topic modeling practitioner is interested in discovering the main thematic information of the two novel datasets in Italian. However, the user does not have prior knowledge on the datasets, therefore does not know which topic model is the most appropriate. Moreover, the user aims to get topics which are coherent and make sense together but which are also diverse and separated from the others. Let us notice that a user could consider a different set of metrics to optimize, by selecting one of the already defined metrics available in OCTIS or by defining novel metrics.

3.1 Evaluation Metrics

We briefly describe the two evaluation metrics (one of topic coherence and one of topic diversity) that we will target as the two objectives of the multi-objective Bayesian optimization. Both metrics need to be maximized.

IRBO (Bianchi et al., 2021a; Terragni et al., 2021b) is a measure of topic diversity (0 for identical topics and 1 for completely different topics). It is based on the Ranked-Biased Overlap measure (Webber et al., 2010). Topics with common words at different rankings are penalized less than topics sharing the same words at the highest ranks.

NPMI (Lau et al., 2014) measures Normalized Pointwise Mutual Information of each pair of words (w_i, w_j) in the 10-top words of each topic. It is a topic coherence measure, that evaluates how much the words in a topic are related to each other.

3.2 Topic Models and Hyper-Parameter Setting

We focus our experiments on four well-known topic models that OCTIS already provides, two of them are considered classical topic models and the others are neural models. In particular, we trained Latent Dirichlet Allocation (Blei et al., 2003, LDA), Non-negative Matrix Factorization (Lee and Seung, 2000, NMF), Embedded Topic Model (Dieng et al., 2020, ETM), Contextualized Topic Models (Bianchi et al., 2021a; Bianchi et al., 2021b, CTM).

Model	Hyper-parameter	Values/Range
All	Number of topics	[5, 100]
LDA	α prior	$[10^{-3}, 10]$
	β prior	$[10^{-3}, 10]$
NMF	Regularization factor	[0, 0.5]
	L1-L2 ratio	[0,1]
	Initialization method	nndsvd, nndsvda, nndsvdar, random
	Regularization	V matrix, H matrix, both
ETM	Activation function	elu, sigmoid, softplus, selu
	Dropout	[0, 0.9]
	Learning rate	$[10^{-3}, 10^{-1}]$
	Number of neurons	$\{100, 200, \ldots, 900, 1000\}$
	Optimizer	adam, sgd, rmsprop
CTM	Activation function	elu, sigmoid, softplus, selu
	Dropout	[0, 0.9]
	Learning rate	$[10^{-3}, 10^{-1}]$
	Momentum	[0, 0.9]
	Number of layers	1, 2, 3, 4, 5
	Number of neurons	$\{100, 200, \ldots, 900, 1000\}$
	Optimizer	adam, sgd, rmsprop

Table 2: Hyper-parameters and ranges.

We summarize the models' hyper-parameters and their corresponding ranges in Table 2. For each model, we optimize the number of topics, ranging from 5 to 100 topics. We select the ranges of the hyper-parameters similarly to previous work (Terragni and Fersini, 2021).

Regarding LDA, we also optimize the hyper-parameters α and β priors that the sparsity of the topics in the documents and sparsity of the words in the topic distributions respectively. These hyper-parameters are set to range between 10^{-3} and 10^{-1} on a logarithmic scale.

The hyper-parameters of NMF are mainly related to the regularization applied to the factorized matrices. The *regularization* hyper-parameter controls if the regularization is applied only to the matrix V, or to the matrix H, or both. The *regularization factor* denotes the constant that multiplies the regularization terms. It ranges between 0 and 0.5 (0 means no regularization). *L1-L2* ratio controls the ratio between L1 and L2-regularization. It ranges between 0 and 1, where 0 corresponds to L2 regularization only, 1 corresponds to L1 regularization only, otherwise it is a combination of the two types. We also optimize the *initialization method* for the two matrices W and H.

Since ETM and CTM are neural models, their hyper-parameters are mainly related to the network architecture. We optimize the *number of neurons* (ranging from 100 to 1000, with a step of 100). For simplicity, each layer has the same number of neurons. We also consider different variants of *activation functions* and *optimizers*. We set the *dropout* to range between 0 and 0.9 and the *learning rate*, that to range between 10^{-3} and 10^{-1}, on a logarithm scale. We fix the batch size to 200 and we adopted an early stopping criterion for determining the convergence of each model.

Moreover, only for CTM we also optimized the *momentum*, ranging between 0 and 0.9, and the number of layers (ranging from 1 to 5). Following (Bianchi et al., 2021b), we use the contextualized document representations derived from SentenceBERT (Reimers and Gurevych, 2019). In particular, we use the pre-trained multilingual Universal Sentence Encoder.[6]

For all the models, we set the remaining parameters to their default values. Finally, we train each model 30 times and consider the median of the 30 evaluations as the evaluation of the function to

[6] Let us notice that there is not a Sentence BERT-like model for Italian. Therefore we used a multilingual one: `distiluse-base-multilingual-cased-v1`.

be optimized. We sample the n initial configurations using the Latin Hypercube Sampling, with n equal to the number of hyperparameters to optimize plus 2 to provide enough configurations for the initial surrogate model to fit. The total number of BO iterations for each model is 125. We use Gaussian Process as the probabilistic surrogate model and the Upper Confidence Bound (UCB) as the acquisition function.

4 Results

In the following, we report the results of the comparative analysis between the considered models on the Italian datasets.

4.1 Quantitative Results

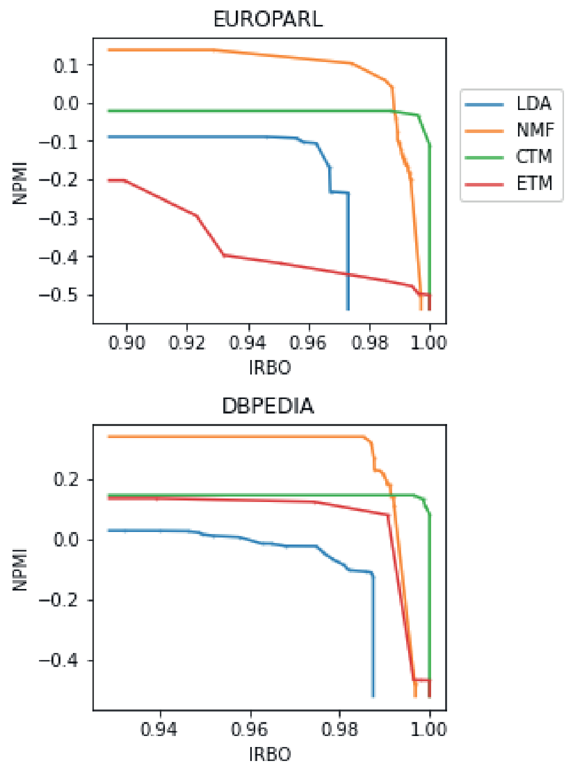


Figure 1: Pareto front of the performance of the considered models for the analyzed Italian datasets.

We jointly consider the results of both objectives by plotting the Pareto frontier of the results of topic diversity and topic coherence. Figure 1 shows the frontier of each model for the pair of metrics (NPMI, IRBO). We can notice that the topic models have similar frontiers in each dataset. The most competitive models are NMF and CTM. In particular, NMF outperforms the others for the topic coherence but gets a lower coherence as the diversity increases. Therefore, CTM is the model to prefer if a user wants to get totally separated topics but good coherence. Instead, LDA and ETM have lower performance than the others. We also noticed from our experiments that the performance of ETM is affected when the documents are shorter (on the Europarl dataset), often originating the phenomenon of mode collapsing, i.e. obtaining all the topics equal to the others.

4.2 Qualitative Results

In Table 3 we report an example of topics discovered by the models. We selected the best hyperparameter configuration discovered by the models with 5 topics and randomly sampled a model run among the 30 runs. Let us notice that, for the sake of simplicity, we have to fix the number of topics here and select a run among the total of 30 runs. Therefore, the qualitative results reported in Table 3 may not reflect the overall results.

We can notice that NMF obtains more coherent and stable topics. CTM and LDA obtain topics that have a higher variance: in particular, CTM discovers a topic (the fourth one, NPMI=-0.51) that lowers the average coherence, while LDA discovers a topic (the second one, NPMI=0.48) that effectively increases the average coherence. On the other hand, the topics discovered by ETM are more stable but have a lower coherence on average. As already observed in previous work (AlSumait et al., 2009; Doogan and Buntine, 2021), obtaining junk or mixed topics is common in topic models and this problem can be addressed by filtering out the topics that are less relevant.

5 Conclusion

In this paper, we presented OCTIS 2.0, the extension of the evaluation framework OCTIS for topic modeling. This tool can now address the problem of estimating the optimal hyper-parameter configurations of different topic models using a multi-objective Bayesian optimization approach. Moreover, we also released two novel datasets in Italian which can be used as benchmark datasets for the Italian topic modeling and NLP communities.

We conducted a simple experimental campaign to show to potentiality of the extended framework. We have seen that using a multi-objective hyperparameter optimization approach allows us not only to identify the best performing model over the oth-

Model	Top words	NPMI
LDA	de album pubblicare italiano the uniti situare fondare università noto	-0.05
	torneo giocare tennis edizione tour atp ambito open categoria cemento	0.48
	film pubblicare the album serie musicale venire statunitense rock band	0.11
	guerra battaglia venire situare statunitense spagnolo partito esercito distretto mondiale	-0.14
	comune campionato squadra abitante calcio regione situare società francese vincere	-0.03
NMF	comune abitante dipartimento regione situare francese alta distretto est grand	0.29
	torneo giocare tennis tour atp open edizione ambito categoria cemento	0.48
	album pubblicare studio the musicale statunitense records singolo cantante rock	0.29
	calciatore ruolo allenatore calcio centrocampista difensore attaccante portiere settembre aprile	0.24
	contea america uniti situare comune censimento designated census place capoluogo	0.39
CTM	album the pubblicare band statunitense singolo brano of musicale rock	0.26
	superare argentino calciatore el buenos maria en svezia situare chiesa	-0.29
	partito battaglia guerra venire politico de linea isola stazione regno	-0.08
	st stella vendetta dollaro robert company ritorno west superiore soggetto	-0.51
	edizione tennis giocare torneo vincere tour campionato maschile disputare squadra	0.18
ETM	sede de italiano fondare nome azienda noto francese compagnia parigi	0.06
	guerra partito battaglia venire nord politico tedesco esercito regno militare	0.03
	torneo situare comune giocare abitante edizione tennis tour regione uniti	-0.10
	film serie the dirigere gioco pubblicare statunitense televisivo venire romanzo	0.07
	album pubblicare campionato squadra musicale the calcio statunitense singolo vincere	-0.12

Table 3: Example of top words of 5 topics for each considered model and the corresponding topic coherence (NPMI).

ers, thus guaranteeing a fairer comparison among different models, but also to empirically discover the relationships between different objectives.

As future work, we aim to extend the framework by considering additional datasets in different and possibly low-resource languages, which require different pre-processing strategies and would allow researchers to investigate the peculiarities of different topic modeling methods.

References

Loulwah AlSumait, Daniel Barbará, James Gentle, and Carlotta Domeniconi. 2009. Topic Significance Ranking of LDA Generative Models. In *Machine Learning and Knowledge Discovery in Databases, European Conference, ECML PKDD 2009*, volume 5781 of *Lecture Notes in Computer Science*, pages 67–82. Springer.

Francesco Archetti and Antonio Candelieri. 2019. *Bayesian Optimization and Data Science*. Springer International Publishing.

Federico Bianchi and Dirk Hovy. 2021. On the gap between adoption and understanding in nlp. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3895–3901.

Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2021a. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021*, pages 759–766. Association for Computational Linguistics.

Federico Bianchi, Silvia Terragni, Dirk Hovy, Debora Nozza, and Elisabetta Fersini. 2021b. Cross-lingual contextualized topic models with zero-shot learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021*, pages 1676–1683. Association for Computational Linguistics.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

David M Blei. 2012. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84.

Jordan L. Boyd-Graber, Yuening Hu, and David M. Mimno. 2017. Applications of topic models. *Found. Trends Inf. Retr.*, 11(2-3):143–296.

Adji Bousso Dieng, Francisco J. R. Ruiz, and David M. Blei. 2020. Topic modeling in embedding spaces. *Trans. Assoc. Comput. Linguistics*, 8:439–453.

Thanh-Nam Doan and Tuan-Anh Hoang. 2021. Benchmarking neural topic models: An empirical study. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages

4363–4368, Online, August. Association for Computational Linguistics.

Caitlin Doogan and Wray L. Buntine. 2021. Topic model or topic twaddle? re-evaluating semantic interpretability measures. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 3824–3848. Association for Computational Linguistics.

Bruno Giovanni Galuzzi, Ilaria Giordani, Antonio Candelieri, Riccardo Perego, and Francesco Archetti. 2020. Hyperparameter optimization for recommender systems through bayesian optimization. *Computational Management Science*, pages 1–21.

Derek Greene and Pádraig Cunningham. 2006. Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering. In *Proceedings of the 23rd International Conference on Machine learning (ICML'06)*, pages 377–384. ACM Press.

Tim Head, Gilles Louppe MechCoder, Iaroslav Shcherbatyi, et al. 2018. scikit-optimize/scikit-optimize: v0. 5.2.

Kirthevasan Kandasamy, Karun Raju Vysyaraju, Willie Neiswanger, Biswajit Paria, Christopher R. Collins, Jeff Schneider, Barnabás Póczos, and Eric P. Xing. 2020. Tuning Hyperparameters without Grad Students: Scalable and Robust Bayesian Optimisation with Dragonfly. *Journal of Machine Learning Research*, 21:81:1–81:27.

Jey Han Lau, David Newman, and Timothy Baldwin. 2014. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014*, pages 530–539.

Daniel D. Lee and H. Sebastian Seung. 2000. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000*, pages 556–562. MIT Press.

Kar Wai Lim and Wray L. Buntine. 2014. Bibliographic analysis with the citation network topic model. In *Proceedings of the Sixth Asian Conference on Machine Learning, ACML 2014*.

Biswajit Paria, Kirthevasan Kandasamy, and Barnabás Póczos. 2019. A Flexible Framework for Multi-Objective Bayesian Optimization using Random Scalarizations. In *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 115 of *Proceedings of Machine Learning Research*, pages 766–776, Tel Aviv, Israel. AUAI Press.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, (EMNLP-IJCNLP)*, pages 3980–3990. Association for Computational Linguistics.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. 2015. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175.

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. 2012. Practical Bayesian Optimization of Machine Learning Algorithms. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems*, pages 2960–2968.

Silvia Terragni and Elisabetta Fersini. 2021. An empirical analysis of topic models: Uncovering the relationships between hyperparameters, document length and performance measures. In *Recent Advances in Natural Language Processing (RANLP)*.

Silvia Terragni, Elisabetta Fersini, and Enza Messina. 2020a. Constrained relational topic models. *Information Sciences*, 512:581 – 594.

Silvia Terragni, Debora Nozza, Elisabetta Fersini, and Messina Enza. 2020b. Which matters most? comparing the impact of concept and document relationships in topic models. In *Proceedings of the First Workshop on Insights from Negative Results in NLP*, pages 32–40.

Silvia Terragni, Elisabetta Fersini, Bruno Giovanni Galuzzi, Pietro Tropeano, and Antonio Candelieri. 2021a. OCTIS: Comparing and Optimizing Topic models is Simple! In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, EACL 2021*, pages 263–270. Association for Computational Linguistics.

Silvia Terragni, Elisabetta Fersini, and Enza Messina. 2021b. Word embedding-based topic similarity measures. In *Natural Language Processing and Information Systems - 26th International Conference on Applications of Natural Language to Information Systems, NLDB 2021*, volume 12801 of *Lecture Notes in Computer Science*, pages 33–45. Springer.

William Webber, Alistair Moffat, and Justin Zobel. 2010. A similarity measure for indefinite rankings. *ACM Trans. Inf. Syst.*, 28(4):20:1–20:38.

How Contextualized Word Embeddings Represent Word Senses

Rocco Tripodi
University of Bologna
rocco.tripodi@unibo.it

Abstract

English. Contextualized embedding models, such as ELMo and BERT, allow the construction of vector representations of lexical items that adapt to the context in which words appear. It was demonstrated that the upper layers of these models capture semantic information. This evidence paved the way for the development of sense representations based on words in context. In this paper, we analyze the vector spaces produced by 11 pre-trained models and evaluate these representations on two tasks. The analysis shows that all these representations contain redundant information. The results show the disadvantage of this aspect.

Italiano. *Modelli come ELMo o BERT consentono di ottenere rappresentazioni vettoriali delle parole che si adattano al contesto in cui queste appaiono. Il fatto che i livelli alti di questi modelli immagazzinino informazione semantica ha portato a sviluppare rappresentazioni di senso basate su parole nel contesto. In questo lavoro analizziamo gli spazi vettoriali prodotti con 11 modelli pre-addestrati e valutiamo le loro prestazioni nel rappresentare i diversi sensi delle parole. Le analisi condotte mostrano che questi modelli contengono informazioni ridondanti. I risultati evidenziano le criticità inerenti a questo aspetto.*

1 Introduction

The introduction of contextualized embedding models, such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019), allows the construction of vector representations of lexical items that adapt to the context in which words appear. It has been shown that the upper layers of these models contain semantic information (Jawahar et al., 2019) and are more diversified than lower layers (Ethayarajh, 2019). These word representations overcame the meaning conflation deficiency that affects static word embedding techniques (Camacho-Collados and Pilehvar, 2018; Tripodi and Pira, 2017), such as *word2vec* (Mikolov et al., 2013) or *GloVe* (Pennington et al., 2014) thanks to the adaptation to the context of use.

The evaluation of these models has been conducted mainly on downstream tasks (Wang et al., 2018; Wang et al., 2019). With extrinsic evaluations, the models are fine-tuned, adapting the vector representations to specific tasks. The resulting vectors are then used as features in classification problems. This hinders a direct evaluation and analysis of the models because the evaluation also takes into account the ability of the classifier to learn the task. A model trained for this kind of task may learn only to discriminate among features that belong to each class with poor generalization.

The interpretability of neural networks is an emerging line of research NLP that aims at analyzing the properties of pre-trained language models (Belinkov and Glass, 2019). Different studies have been conducted in recent years to discover what kind of linguistic information is stored in large neural language models. Many of them are focused on syntax (Hewitt and Manning, 2019; Jawahar et al., 2019) and attention (Michel et al., 2019; Kovaleva et al., 2019). For what concerns semantics, the majority of the studies focus on common knowledge (Petroni et al., 2019) and inference and role-based event prediction (Ettinger, 2020). Only a few of them have been devoted to lexical semantics, for example, Reif et al. (2019) show how different representations of the

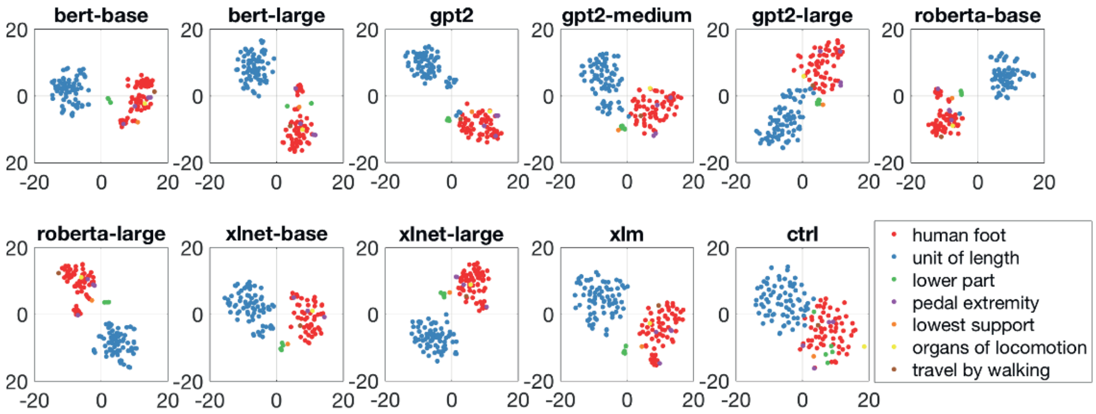


Figure 1: t-SNE representations for the word *foot* in SemCor, grouped by sense.

same lexical form tend to cluster according to their sense.

In this work, we propose an in-depth analysis of the properties of the vector spaces induced by different embedding models and an evaluation of their word representations. We present how the properties of the vector space contribute to the success of the models in two tasks: sense induction and word sense disambiguation. In fact, even if contextualized models do not create one representation per word sense (Ethayarajh, 2019), their contextualization create similar representations for the same word sense that can be easily clustered.

2 Related Work

Given the success (and the opacity) of contextualized embedding models, many works have been proposed to analyze their inner representations. These analyses are based on *probing tasks* (Conneau et al., 2018) that aim at measuring how the information extracted from a pre-trained model is useful to represent linguistic structures. Probing tasks involve training a diagnostic classifier to determine if it encodes desired features. Tenney et al. (2019) discovered that specific BERT's layers are more suited for representing information useful to solve specific tasks and that the ordering of its layers resembles the ordering of a traditional NLP pipeline: POS tagging, parsing, NER, semantic role labeling, and coreference resolution. Hewitt and Manning (2019) evaluated whether syntax trees are embedded in a linear transformation of a neural network's word representation space. Hewitt and Liang (2019) raised the problem of interpreting the results derived from probing analysis. In fact, it is difficult to understand whether high accuracy values are due to the representation itself or, instead, they are the result of the ability to learn a specific task during training.

Our work is more in line with works that try to find general properties of the representations generated by different contextualized models. For example, Mimno and Thompson (2017) demonstrated that the vector space produced by a static embedding model is concentrated in a narrow cone and that its concentration depends on the ratio of positive and negative examples. Mu and Viswanath (2018) explored this analysis further, demonstrating that the embedding vectors share the same common vector and have the same main direction. Ethayarajh (2019) demonstrated how upper layers of a contextualizing model produce more contextualized representations. We built on top of these works analyzing the vector space generated by contextualized models and evaluating them.

3 Construction of the Vector Spaces

We used SemCor (Miller et al., 1993) as reference corpus for our work. This choice is motivated by the fact that it is the largest dataset manually annotated with sense information and it is commonly used as training set for word sense disambiguation. It contains 352 documents whose content words (about $226,000$) have been annotated with WordNet (Miller, 1995) senses. In total there are $33,341$ unique senses distributed over $22,417$ different words. The sense distribution in this corpus is very skewed, and follows a power law (Kilgarriff, 2004). This makes the identification of senses challenging. The dataset is also difficult due to the

Model	training data	vocab. size	n. param.	vec. dim.	objective
$\text{BERT}_{\text{base}}$ (Devlin et al., 2019)	16GB	30K	110M	768	masked language model and next sentence prediction
$\text{BERT}_{\text{large}}$ (Devlin et al., 2019)	16GB	30K	340M	1024	masked language model and next sentence prediction
$\text{GPT-2}_{\text{base}}$ (Radford et al., 2019)	40GB	50K	117M	768	language model
$\text{GPT-2}_{\text{medium}}$ (Radford et al., 2019)	40GB	50K	345M	1024	language model
$\text{GPT-2}_{\text{large}}$ (Radford et al., 2019)	40GB	50K	774M	1280	language model
$\text{RoBERTa}_{\text{base}}$ (Liu et al., 2019)	160GB	50K	125M	768	masked language model
$\text{RoBERTa}_{\text{large}}$ (Liu et al., 2019)	160GB	50K	355M	1024	masked language model
$\text{XLNet}_{\text{base}}$ (Yang et al., 2019)	126GB	32K	110M	768	bidirectional language model
$\text{XLNet}_{\text{large}}$ (Yang et al., 2019)	126GB	32K	340M	1024	bidirectional language model
$\text{XLM}_{\text{english}}$	16GB	30K	665M	2048	language model
CTRL (Keskar et al., 2019)	140GB	250K	1.63B	1280	conditional transformer language model

Table 1: Statistics and hyperparameters of the models.

Model	AvgNorm	MeanVecNorm(A)	MeanVecNorm($\hat{A}$)	avg.MEV	avg.IntSim	avg.ExtSim
$\text{BERT}_{\text{base}}$	25.78 ± 1.28	17.94	17.84	0.43 ± 0.18	0.74 ± 0.05	0.69 ± 0.06
$\text{BERT}_{\text{large}}$	20.83 ± 2.51	12.43	11.58	0.38 ± 0.18	0.66 ± 0.08	0.59 ± 0.08
$\text{GPT-2}_{\text{base}}$	125.13 ± 10.25	91.46	90.99	0.46 ± 0.18	0.79 ± 0.05	0.76 ± 0.05
$\text{GPT-2}_{\text{medium}}$	427.45 ± 38.78	371.86	360.36	0.51 ± 0.18	0.85 ± 0.03	0.84 ± 0.03
$\text{GPT-2}_{\text{large}}$	290.29 ± 38.56	226.39	212.97	0.43 ± 0.18	0.75 ± 0.05	0.72 ± 0.05
$\text{RoBERTa}_{\text{base}}$	25.78 ± 0.56	22.17	22.25	0.51 ± 0.17	0.87 ± 0.02	0.85 ± 0.03
$\text{RoBERTa}_{\text{large}}$	31.47 ± 0.65	26.99	27.04	0.52 ± 0.18	0.88 ± 0.02	0.84 ± 0.03
$\text{XLNet}_{\text{base}}$	47.68 ± 0.66	43.28	43.26	0.53 ± 0.17	0.88 ± 0.01	0.87 ± 0.02
$\text{XLNet}_{\text{large}}$	28.27 ± 1.42	19.56	19.68	0.38 ± 0.17	0.66 ± 0.04	0.62 ± 0.05
$\text{XLM}_{\text{english}}$	44.92 ± 2.61	37.13	36.7	0.45 ± 0.18	0.79 ± 0.03	0.77 ± 0.03
CTRL	4443.62 ± 351.98	3927.86	3879.56	0.49 ± 0.18	0.84 ± 0.02	0.83 ± 0.02

Table 2: Detailed description of the embedding space produced with each model.

fine granularity of WordNet (Navigli, 2006).

To construct the vector space A from SemCor we collected all the senses S_i of a word w_i and for each sense $s_j \in S_i$ we recovered the sentences $\{Sent_1^{w_i s_j}, Sent_2^{w_i s_j}, ..., Sent_n^{w_i s_j}\}$ in which this particular sense occurs. These sentences are then fed into a pre-trained model and the token embedding representations of word w_i, $\{e_1^{w_i s_j}, e_2^{w_i s_j}, ..., e_n^{w_i s_j}\}$, are extracted from the last hidden layer. This operation is repeated for all the senses in S_i, and for all the tagged words in the vocabulary, V. The vector space corresponds to all the representations of the words in V.

A t-SNE visualization of the different embeddings in SemCor for the word *foot* is presented in Figure 1. In this Figure, we can see that the three main senses of *foot* (i.e., human foot, unit of length and lower part) occupy a definite position in the vector space, suggesting that the models are able to produce specific representations for the different senses of a word and that they lie on defined subspaces. In this work we want to test to what extent this feature is present in language models.

Implementations details The pre-trained models used in this study are: two BERT (Devlin et al., 2019) models, *base cased* (12-layer, 768-hidden, 12-heads, 110M parameters) and *large cased* (24-layer, 1024-hidden, 16-heads, 340M parameters); three GPT-2 (Radford et al., 2019) models, *base* (12-layer, 768-hidden, 12-heads, 117M parameters), *medium* (24-layer, 1024-hidden, 16-heads, 345M parameters) and *large* (36-layer, 1280-hidden, 20-heads, 774M parameters); two RoBERTa (Liu et al., 2019) models, *base* (12-layer, 768-hidden, 12-heads, 125M parameters) and *large* (24-layer, 1024-hidden, 16-heads, 355M parameters); two XLNet (Yang et al., 2019) models, *base* (12-layer, 768-hidden, 12-heads, 110M parameters) and *large* (24-layer, 1024-hidden, 16-heads, 340M parameters); one XLM (Lample et al., 2019) model (12-layer, 2048-hidden, 16-heads) and one CTRL (Keskar et al., 2019) model (48-layer, 1280-hidden, 16-heads, $1.6B$ parameters). The main features of these models are summarized in Table 1. We averaged the embeddings of sub-tokens to obtain token-level representations.

We used the transformers library (Wolf et al., 2019).

3.1 Analysis

The first objective of this work is to analyze the vector space produced with the models. This analysis is aimed at investigating the properties of the contextualized vectors. A detailed description of the embedding spaces constructed with the pre-trained models is presented in Table 2. We computed the norm for all the vectors in the vector space A, and averaged them:

$$AvgNorm = \frac{1}{|A|} \sum_{i=1}^{|A|} \|e_i\|_2. \quad (1)$$

This measure gives us an intuition on how diverse the semantic space constructed with the different models is. In fact, we can see that the magnitude of the vectors constructed with BERT, RoBERTa, XLNet, and XLM is low while those of GPT-2 and CTRL are very high.

We computed also the norm of the vector resulting in averaging all the vectors in the semantic space V, as:

$$MeanVecNorm = \left\| \frac{1}{|A|} \sum_{i=1}^{|A|} e_i \right\|_2. \quad (2)$$

All the semantic spaces have non-zero mean and the mean norm is high. This result suggests that the vectors contain redundant information and share a common nonzero vector. This is not only because the vector space contains representations of the same sense. In fact, if we create a new semantic space, $\hat{A}$, averaging all the representations of the same word sense, the $MeanVecNorm$ of this space is still high for all the models.

We used the Maximum Explainable Variance (MEV) for the representations of each word in V. This measure corresponds to the proportion of the variance in the embeddings that can be explained by their first principal components and was computed as:

$$MEV(w) = \frac{\sigma_1^2}{\sum_i \sigma_i^2}. \quad (3)$$

where $\sigma_i^2 1$ is the first principal component of the vector space A. It can give an upper bound on how contextualized representations can be replaced by a static embedding (Ethayarajh, 2019). The model with the lowest MEV is $\text{BERT}_{\text{large}}$ and $\text{XLNet}_{\text{large}}$.

The other measures that we used for the evaluation of the vector space are based on the very notion of a cluster, which imposes that the data points inside a cluster must satisfy two conditions: internal similarity and external dissimilarity (Pelillo, 2009). To this end, we used the senses of each word in the vocabulary of SemCor as clusters and extracted the corresponding vectors from V. We

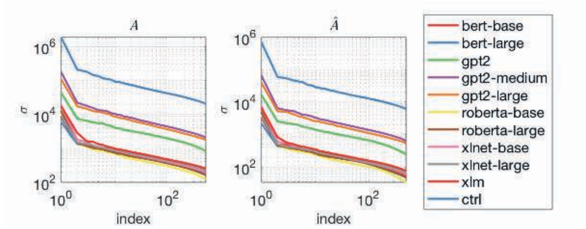


Figure 2: The first 500 principal components computed on A and $\hat{A}$.

then computed the *internal similarity* of a cluster, c, as:

$$IntSim(c) = \frac{1}{n^2 - n} \sum_j \sum_{k \neq j} cos(e_j, e_k), \quad (4)$$

where n is the number of data points in the cluster. We computed also the *external similarity* of a cluster c by computing the cosine similarity among each point in c and all the points in the subspace S induced by the senses of a word that has c as one of its senses:

$$ExtSim(c) = \frac{1}{n \cdot m} \sum_{j=1}^{n} \sum_{k=1}^{m} cos(e_j, e_k), \quad (5)$$

where m is the total number of data points in the subspace S (excluding those in c) and n is the number of points in the cluster c. Our hypothesis is that good representations should have high internal similarity and low external similarity and that the difference between them should be high.

As it can be seen from Table 2 the internal similarity is higher than the external for all the models. Despite this, the scores are in a wide range. The lowest $IntSim$ is given by $\text{BERT}_{\text{large}}$ and the highest by $\text{RoBERTa}_{\text{large}}$ and $\text{XLNet}_{\text{base}}$. The lowest $ExtSim$ is given by $\text{BERT}_{\text{large}}$ and the highest by $\text{XLNet}_{\text{base}}$. The largest difference between the two measures is given by $\text{BERT}_{\text{large}}$. $\text{RoBERTa}_{\text{large}}$ gives has also a large gap between the two measures, furthermore, their standard deviation is very low. As we will see in Section 4 these last two models perform better than others in clustering and classification tasks.

4 Evaluation

Sense Induction This task is aimed at understanding if representations belonging to different senses can be separated using an unsupervised approach. We hypothesize that a good contextualization process should produce more discriminative

model	k-means					dominant-set				
	N	V	A	R	All	N	V	A	R	All
$\text{BERT}_{\text{base}}$	57.2	50.6	56.2	62.0	54.9 ± 14.8	55.7	45.3	51.7	45.8	51.0 ± 17.5
$\text{BERT}_{\text{large}}$	**59.3**	51.9	56.9	59.0	56.2 ± 15.3	53.4	42.6	46.8	39.9	47.8 ± 17.1
$\text{GPT-2}_{\text{base}}$	54.1	48.3	55.6	56.8	$\underline{52.3}\pm14.7$	54.3	45.3	50.2	46.3	50.1 ± 17.2
$\text{GPT-2}_{\text{medium}}$	53.9	49.1	56.2	59.8	52.8 ± 14.5	59.7	49.8	58.7	54.8	56.0 ± 18.8
$\text{GPT-2}_{\text{large}}$	53.8	49.4	58.1	58.8	53.0 ± 14.8	50.2	44.1	46.1	44.1	47.1 ± 16.0
$\text{RoBERTa}_{\text{base}}$	56.4	51.4	56.7	59.7	54.8 ± 14.7	65.3	55.1	64.8	61.4	61.6 ± 19.2
$\text{RoBERTa}_{\text{large}}$	58.5	**53.0**	**58.6**	**62.7**	$\mathbf{56.7}\pm14.9$	66.7	**56.6**	66.3	**64.2**	$\mathbf{63.2}\pm19.3$
$\text{XLNet}_{\text{base}}$	54.2	49.1	53.8	56.8	52.2 ± 14.4	**67.2**	55.0	**68.7**	63.8	$\underline{62.7}\pm20.7$
$\text{XLNet}_{\text{large}}$	57.6	52.5	57.9	60.8	$\underline{55.9}\pm14.4$	51.0	44.8	47.5	40.9	$\underline{47.6}\pm15.0$
$\text{XLM}_{\text{english}}$	56.3	50.1	56.5	62.1	54.3 ± 15.1	60.4	51.3	59.5	55.9	57.0 ± 18.1
CTRL	53.8	47.0	56.5	57.4	51.9 ± 15.4	60.4	49.4	61.7	56.3	56.8 ± 19.2

Table 3: Results (as average accuracy) on clustering divided by algorithm and part of speech: nouns (N), verbs (V), adjectives (A), adverbs (R) and on the concatenations of all datasets (All).

representations that can be easily identified by a clustering algorithm.

We used the sense clusters extracted from SemCor as ground truth for this experiment (see Section 3) and grouped them if they are senses of the same word (with a given part of speech). We retained only the groups that have at least 20 data points and we discarded also monosemous words for the evaluation on k-means. The resulting datasets consist of 1871 (entire) and 1499 (without monosemous words) sub-datasets with $141,074$ and $116,019$ data points in total, respectively. We computed the accuracy on each sub-dataset computing the number of data points that have been clustered correctly and averaged the results to measure the performance of each model.

The first algorithm is k-means (Lloyd, 1982). It is a partitioning, iterative algorithm whose objective is to minimize the sum of point-to-centroid distances, summed over all k clusters. We used the k-means++ heuristic (Arthur and Vassilvitskii, 2007) and the cosine distance metric to determine distances. We selected this algorithm because it is simple, non-parametric, and is widely used. It is important to notice that k-means requires the number of clusters to extract, for this reason, we restricted the evaluation only to ambiguous words.

The second algorithm used is *dominant-set* (Pavan and Pelillo, 2007). It is a graph-based algorithm that extracts compact structures from graphs generalizing the notion of maximal clique defined on unweighted graphs to edge-weighted graphs. We selected this algorithm because it is non-parametric, requires only the adjacency matrix of a weighted graph as input, and, more importantly, does not require the number of clusters to extract. The clusters are extracted from the graph sequentially using a peel-off strategy. This feature allows us to include in the evaluation also unambiguous words and to see if their representations are grouped into a single cluster or partitioned into different ones. We used cosine similarity to weigh the edges of the input graph.

The results of this evaluation are presented in Table 3. RoBERTa and BERT have the overall best performances on this task using both algorithms. In particular, $\text{RoBERTa}_{\text{large}}$ performs consistently well on all parts of speech and across algorithms, while other models perform well only in combination with one of the two algorithms. This is presumably owing to the big gap between the internal and the external similarity produced by this model, as explained in Section 3.1.

This evaluation tends to confirm the claim that larger versions of the same model achieve better results. From Table 3, we can also see that the models have more difficulties in identifying the different senses of verbs, while nouns and adverbs have higher results. This is probably due to the different distribution of these word classes in the training sets of the models and WordNet's fine-granularity. The performances of the models with dominant-set are surprisingly high, considering that the setting of this experiment is completely unsupervised. Furthermore, this algorithm is conceived to extract compact clusters and this feature could drive it to over partition the vector space of monosemous words. Instead, the results suggest the opposite: that the models are able to produce representations with high internal similarity, positioning their representations on a defined sub-space.

Word Sense Disambiguation We used the method proposed in Peters et al. (2018) to create

Model	S2			S3			SE07			SE13			SE15			All		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
BERT$_{\text{base}}$	80.6	67.9	73.7	77.2	68.8	72.8	66.4	63.1	64.7	74.4	62.7	68.1	78.3	68.8	73.2	77.0	66.8	71.5
BERT$_{\text{large}}$	81.2	68.4	74.3	80.3	71.5	**75.6**	68.5	65.1	66.7	75.8	63.9	69.3	79.7	70.1	74.6	77.9	67.5	72.3
GPT-2$_{\text{base}}$	75.6	63.7	69.1	71.5	63.7	67.4	59.3	56.3	57.7	71.8	60.5	65.7	74.4	65.4	69.6	72.4	62.8	67.2
GPT-2$_{\text{medium}}$	76.5	64.5	70.0	72.9	65.0	68.7	62.0	58.9	60.4	74.0	62.3	67.7	76.6	67.3	71.7	74.0	64.2	68.8
GPT-2$_{\text{large}}$	76.4	64.4	69.9	72.1	64.2	67.9	61.8	58.7	60.2	72.8	61.4	66.6	75.6	66.3	70.7	73.4	63.6	68.1
RoBERTa$_{\text{base}}$	82.0	69.1	**75.0**	79.4	70.7	74.8	66.7	63.3	64.9	75.5	63.7	69.1	79.5	69.9	74.4	78.5	68.0	72.9
RoBERTa$_{\text{large}}$	82.0	69.1	**75.0**	80.0	71.2	75.4	70.6	67.0	**68.8**	77.1	65.0	**70.5**	81.0	71.1	**75.7**	79.4	68.9	**73.8**
XLNet$_{\text{base}}$	78.8	65.8	71.7	76.2	67.4	71.5	67.3	63.7	65.5	70.7	58.3	63.9	77.5	67.1	71.9	75.4	64.6	69.5
XLNet$_{\text{large}}$	80.6	67.9	73.7	78.7	70.1	74.2	67.6	64.2	65.8	75.3	63.5	68.9	80.6	70.8	75.4	78.0	67.7	72.5
CTRL	73.4	61.9	67.1	70.1	62.5	66.1	54.2	51.4	52.8	68.2	57.5	62.4	72.3	63.5	67.6	69.9	60.6	64.9

Table 4: Results indicating precision (P), recall (R) and F1 on each dataset and on their concatenation (All). All the results are computed using $\hat{A}$ as vector space.

sense vectors from contextualized word vectors. This method consists in averaging all the representations of a given sense. The resulting vector space corresponds to $\hat{A}$ (see Section 3.1). We evaluated the generated vectors on a standard benchmark (Raganato et al., 2017) for WSD. It consists of five datasets that were unified to the same WordNet version: Senseval-2 (S2), Senseval-3 (S3), SemEval-2007 (S7), SemEval-2013 and SemEval-2015, having in total $10,619$ target words.

The identification of word senses is conducted by feeding the entire texts of the datasets into a pre-trained model and extracting, for each target word w_i, its embedding representation $e_k^{w_i}$ as was done for the construction of the semantic space. Once these representations are available, we compute the cosine similarities among $e_k^{w_i}$ and the embeddings in $\hat{A}$ constructed with the same model and selected the sense with the highest similarity. We did not use more sophisticated models such as WSD-games (Tripodi and Navigli, 2019; Tripodi et al., 2016) because we wanted to keep the evaluation as simple as possible as not to influence the evaluation of the results.

The results of this evaluation are presented in Table 4. The first trend that emerges from the results is the big gap between *precision* and *recall*. This is due to the absence of many senses in our training set. We did not want to use back-off strategies or other techniques usually employed in the WSD literature, to not influence the performances and the analysis of the results. Despite the simplicity of the approach, it performs surprisingly well. In particular, BERT, RoBERTa, and XLNet (three bidirectional models) have very high results. The low performances of CTRL are probably due to its large vocabulary and to its objective, designed to solve different tasks.

5 Conclusion and Future Work

We conducted an extensive analysis of the semantic capabilities of contextualized embedding models. We analyzed the vector space constructed using pre-trained models and found that their vectors contain redundant information and that their first two principal components are dominant.

The results on sense induction are promising. They demonstrated the effectiveness of contextualized embeddings to capture semantic information. We did not find higher performances from more complex models, rather, we found that RoBERTa, a model that was developed by simplifying a more complex model, BERT, was one of the best performers. Neither the dimension of the hidden layers, the size of the training data, nor the size of the vocabulary seems to play a big role in modeling semantics. As stated in previous works, inserting an anisotropy penalty to the objective function of the models could improve directly the representations. We also noticed that, even if BERT models and XLNet have different objectives and are trained on different data, they have similar performances. It emerged that these models are less redundant than others.

The conclusion that we can draw from our analysis and evaluation is that pre-trained language models can capture lexical-semantic information and that unsupervised models can be used to distinguish among their representations. On the other hand, these representations are redundant and anisotropic. We hypothesize that reducing these aspects can lead to better representations. This operation can be carried out *post-hoc* but we think that training new models keeping this point in mind could lead to the development of better models.

References

David Arthur and Sergei Vassilvitskii. 2007. k-means++: the advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 1027–1035.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72, March.

José Camacho-Collados and Mohammad Taher Pilehvar. 2018. From word to sense embeddings: A survey on vector representations of meaning. *J. Artif. Intell. Res.*, 63:743–788.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia, July. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China, November. Association for Computational Linguistics.

Allyson Ettinger. 2020. What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China, November. Association for Computational Linguistics.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July. Association for Computational Linguistics.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Adam Kilgarriff. 2004. How dominant is the commonest sense of a word? In Petr Sojka, Ivan Kopeček, and Karel Pala, editors, *Text, Speech and Dialogue*, pages 103–111, Berlin, Heidelberg. Springer Berlin Heidelberg.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4365–4374, Hong Kong, China, November. Association for Computational Linguistics.

Guillaume Lample, Alexandre Sablayrolles, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2019. Large memory layers with product keys. *arXiv preprint arXiv:1907.05242*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Stuart P. Lloyd. 1982. Least squares quantization in PCM. *IEEE Trans. Information Theory*, 28(2):129–136.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 14014–14024.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A semantic concordance. In *HUMAN LANGUAGE TECHNOLOGY: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November.

David Mimno and Laure Thompson. 2017. The strange geometry of skip-gram with negative sampling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2873–2878, Copenhagen, Denmark, September. Association for Computational Linguistics.

Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-top: Simple and effective postprocessing for word representations. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Roberto Navigli. 2006. Meaningful clustering of senses helps boost word sense disambiguation performance. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, pages 105–112, Stroudsburg, PA, USA. Association for Computational Linguistics.

Massimiliano Pavan and Marcello Pelillo. 2007. Dominant sets and pairwise clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(1):167–172.

Marcello Pelillo. 2009. What is a cluster? perspectives from game theory. In *Proc. of the NIPS Workshop on Clustering Theory*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China, November. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110, Valencia, Spain, April. Association for Computational Linguistics.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B. Viégas, Andy Coenen, Adam Pearce, and Been Kim. 2019. Visualizing and measuring the geometry of BERT. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 8592–8600.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July. Association for Computational Linguistics.

Rocco Tripodi and Roberto Navigli. 2019. Game theory meets embeddings: a unified framework for word sense disambiguation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 88–99, Hong Kong, China, November. Association for Computational Linguistics.

Rocco Tripodi and Stefano Li Pira. 2017. Analysis of italian word embeddings. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017), Rome, Italy, December 11-13, 2017*.

Rocco Tripodi, Sebastiano Vascon, and Marcello Pelillo. 2016. Context aware nonnegative matrix factorization clustering. In *23rd International Conference on Pattern Recognition, ICPR 2016, Cancún, Mexico, December 4-8, 2016*, pages 1719–1724.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, November. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 3261–3275.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

ConteCorpus: An Analysis of People Response to Institutional Communications During the Pandemic

Viviana Ventura, Elisabetta Jezek
Department of Humanities
University of Pavia, Pavia, Italy
viviana.ventura01@universitadipavia.it, jezek@unipv.it

Abstract

The study of institutional communication related to the pandemic, and to the population's response to it, is of great relevance today. The Italian spokesperson for communication regarding the pandemic has been, during the year 2020, the former Prime Minister Giuseppe Conte. We retrieved 4,860,395 comments from his Facebook official page and built the ConteCorpus, a new Italian resource annotated in CoNLL-U format. A first aim of the research was to evaluate the performance of the model used to annotate the corpus. Models trained on social media texts are usually not very generalizable. Nevertheless, the results of the evaluation were good, especially in parsing metrics, and showed that a parser trained on Twitter data can be successfully applied to Facebook data. A second aim of the research was to provide an overall view of the content of such a large corpus; for this purpose, topic modeling was conducted, training an LDA model. The model generated 5 topics that cover different aspects linked to the pandemic emergency, from economic to political issues. Through the topic modeling we investigated which topics are prevalent on particular days.

1 Introduction

During the year 2020, the Prime Minister Giuseppe Conte has played a major role in institutional communication, particularly in communication regarding the policies undertaken to manage the health emergency. We assumed that interesting content from the point of view of the response of the population to institutional communications regarding the pandemic would have been found on his social media profiles. Therefore, we created ConteCorpus,[1] retrieving more than 4 million comments from his Facebook page[2] starting from January 2020 until December 2020, and we annotated it in CoNLL-U format[3].

A first aim of the research was to evaluate the performance of the model used to annotate the dataset. Models trained on social media texts usually are poorly generalizable even on text retrieved from the same social media, therefore we wanted to test the performance on Facebook texts of a model trained on Twitter texts. In order to evaluate the model, we created a gold standard by extracting 1,000 sentences from the ConteCorpus and manually revising them.

A second aim of the research was to provide an overall view of this large corpus. For this purpose we performed a Topic Modeling. We trained a LDA model sampling 10% of the ConteCorpus. The LDA model generated 5 topics related to different aspects of the pandemic emergency. The model was used to see which topics were the most relevant before and after the announcement of the first and the second period of restrictions adopted to fight the pandemic in Italy.

The paper is structured as follows: we first review the relevant literature for our research (section 2), then we describe the data collection and the creation of the corpus (section 3). In section 4, we describe the evaluation we performed of the model we used to annotate the corpus in CoNLL-U format, and in section 5 we report the results of the topic modeling experiment. In section 6 we provide some concluding observations.

[1] https://github.com/Viviana-dev/Conte_Corpus
[2] https://www.facebook.com/GiuseppeConte64/
[3] https://universaldependencies.org/format.html

	January	February	March	April	May	June	July	August	September	October	November	December	Tot
Post	48	59	48	45	26	44	61	24	43	75	33	28	534
Comment	115,971	154,266	681,221	775,972	361,179	335,772	449,913	190,777	260,237	666,126	441,822	427,139	4,860,395

Table 1. Number of posts and comments retrieved for each month.

2 State of the Art

Since the beginning of the health emergency, there has been a proliferation of computational analyses that exploit data extracted from social media. These data are considered relevant as they allow us to generalize about human social and linguistic behavior, especially regarding the pandemic event. Among the tasks that have been conducted on data drawn from social media in this period, sentiment analysis, emotion profiling and topic modeling are the most common (Gagliardi et al., 2020; Tamburini, 2020; Vitale et al., 2020; Stella et al., 2020a; Stella et al., 2020b; Stella et al., 2021; De Santis et al., 2020; Sciandra, 2020; Trevisan et al., 2021; Gozzi et al., 2020; Kruspe et al., 2020; Hussain et al., 2021; Chakraborty et al., 2020; Nemes e Kiss, 2020; Jelodar et al., 2021; Lamsal, 2020; Duong et al., 2021; Gupta et al., 2021; Sullivan et al., 2021; Su et al., 2020; Garcia et Berton, 2021; Ahmed et al., 2020).

In particular, Topic Modeling aims at finding hidden semantic structures within the texts and to model them into concepts. The unsupervised clustering technique LDA (Latent Dirichlet Allocation), developed by Blei (2003), has been used extensively in analyses conducted on social media data during the pandemic (Dashtian et Murthy, 2021; Feng et Zhou, 2020; Ordun et al., 2020; Wang et al., 2020; Kabir et Mandria, 2020; Amara et al., 2020; Abd-Alzaraq et al., 2020; Naseem et al., 2021; Low et al. 2020, Andreadis et al., 2021). LDA is a statistical model that represents each document in a corpus as a probabilistic distribution over latent topics and each topic as a probabilistic distribution over words. A topic has a probability of generating various words, where the words are all the observed words in the corpus. Thus, the terms in the set of documents are used to discover hidden topics in a large corpus.

As is well known, the language of the web is characterized by deviation from the standard language that challenges the use of NLP tools. Several classifications have been proposed to label the nature of web and social media language. In general, the labels aim to define a variety of language that is diaphasically low and at an indefinite point on the diamesic axis, e.g., "netspeak" (Crystal, 2001). Web and social media language is characterized by little planning in text structure and a greater propensity for parataxis, absence of revision and punctuation, abrupt interruption of periods, and an imitation of the continuous flow of speech (Fiorentino, 2013). Although some persistent traits of web and social media language can be described, it does not constitute a single variety of language from a sociolinguistic perspective (Fiorentino, 2013). This poses a double challenge in the use of NLP tools. First, because the tools are calibrated to standard language variety resources. Secondly, even if we created models that are better suited to web and social media languages, they would not be generalizable to every language variety on the web (Sanguinetti et al., 2018).

3 ConteCorpus Construction

3.1 Data Collection

We have downloaded 4,860,395 comments and 534 posts published during the year 2020 on Giuseppe Conte's Facebook official profile. We made call to any 2020 post ID of Giuseppe Conte's official page to retrieve text, object id, and created time of comments. The calls to the Facebook API Graph[4] were made month to month in the same fashion. Nevertheless, as Table 1 shows, a larger amount of data has been retrieved in the month of March, April, and October. In the same period in Italy the more restrictive measures to fight pandemic were taken by the government.

3.2 Processing with the Neural Pipeline Stanza

After the data collection, we processed the data with the Neural Pipeline Stanza[5] to enrich the texts with some annotations. Stanza is an open-source Python NLP toolkit, which "features a language-agnostic fully neural pipeline for text analysis, including tokenization, multiword token expansion, lemmatization, part-of-speech and morphological feature tagging, dependency parsing, and named entity" (Qi et al., 2020). The kit supports more than 77 human languages and uses the

[4] https://developers.facebook.com/docs/graph-api?locale=it_IT

[5]https://stanfordnlp.github.io/stanza/

	Tokens	Words	UPOS	XPOS	UFEATS	AllTags	Lemmas	UAS	LAS	CLAS	MLAS	BLEX
Precision	97.71	97.53	95.84	95.83	95.71	95.12	95.98	86.17	83.10	78.59	76.25	77.39
Recall	94.65	94.44	92.81	92.80	92.68	92.11	92.94	83.44	80.47	76.83	74.54	75.65

Table 2. Performance of Stanza's UD pre-trained model tested on the test set of ConteCorpus.

	Tokens	Words	UPOS	XPOS	UFeats	AllTags	Lemmas	UAS	LAS	CLAS	MLAS	BLEX
PoSTWITA-UD	99.71	99.46	96.19	96.04	96.28	95.01	97.7	82.67	78.27	72.2	68.55	70.35
ConteCorpus	96.15	95.96	94.30	94.29	94.17	93.59	94.44	84.78	81.76	77.70	75.38	76.59

Table 3. Performance of Stanza's UD pre-trained model tested on official test set of PoSTWITA-UD and on test set of ConteCorpus. The scores shown are calculated using the F-measure.

formalism Universal Dependencies[6] Knowing the difficulties of annotating non standard texts such as those derived from social media, we chose to use this pipeline because the evaluation of its models found that Stanza neural language agnostic architecture "adapts well to text of different genres […] achieving state-of-the-art or competitive performance at each step of the pipeline" (Qi et al., 2020). Moreover, models that can be downloaded from Stanza have been trained each on a single language and on a specific text genre dataset. We chose to download the model trained on PoSTWITA-UD.[7] PoSTWITA-UD is an Italian Twitter treebank in Universal Dependencies (Sanguinetti et al., 2018). Although the language of social media is very peculiar and changes from one social media to another and from groups to groups (Fiorentino, 2013), we thought that the model downloadable from Stanza - trained on this dataset - could be generalizable to our data, being in-domain. Moreover, Sanguinetti et al. (2018) have added customized tags to the UD scheme to deal with some social media peculiar phenomena: "discourse:emo" for emojis and emoticons, and "parataxis:hashtag" for hashtags. They tagged the link found in some sentences as "dep" (unspecified relation) and used the "upos" (universal part-of-speech) tag "SYM" (symbol) for hashtags and emojis. Additionally, they manually inserted the lemma of non-standard word forms not recognized by the lemmatizer (Sanguinetti et al., 2018).

We processed the data divided in 12 packages; each correspond to one month data. We used every processor of the pipeline, besides the Named Entity Recognition module (TokenizeProcessor, POSProcessor, LemmaProcessor, DepparseProcessor). We personalized the model in order not to split the sentences,[8] forcing the TokenizeProcessor to consider each comment as a sentence. Furthermore, we added two metadata to each sentence: one refers to the id of the post from which the comment was retrieved, and the other is the creation time of the comment. The aim is to make it easier to retrieve the comments from the corpus by their created time or post id if one needs to analyze a particular period of time or a particular post.

4 End-to-End Evaluation

4.1 Construction of the Gold Standard

We built a gold standard with a dual purpose: to evaluate the performance of the model on this new collection of social media texts, and to create a standard that can be used for future training and testing. We randomly selected 83 sentences from each file of the corpus annotated automatically (one file is composed of one-month comments), and manually revised the 1,000 sentences collected. The manual revision has followed the principle that what is understandable by a human would be correct.

4.2 Evaluation with CoNLL 2018 UD Shared Task Official Evaluation Script

To perform the evaluation, we used CoNLL 2018 UD shared task official evaluation script.[9] Table 2 shows the scores of evaluation metrics resulting from the performance of Stanza model on the test set of the ConteCorpus. Table 3 compares the scores of evaluation metrics resulting from the performance of Stanza model on the test set of PoSTWITA-UD and the ConteCorpus. The first two columns are the scores on metrics that evaluate segmentation. The row called UPOS shows the

[6] Universal Dependencies (UD) is a "framework for consistent annotation of grammar (parts of speech, morphological features, and syntactic dependencies) across different human languages" (https://universaldependencies.org/).

[7] https://universaldependencies.org/treebanks/it_postwita/index.html

[8] Sentence segmentation and tokenization are jointly performed by the TokenizeProcessor (Qi et al., 2020).

[9] https://universaldependencies.org/conll18/evaluation.html

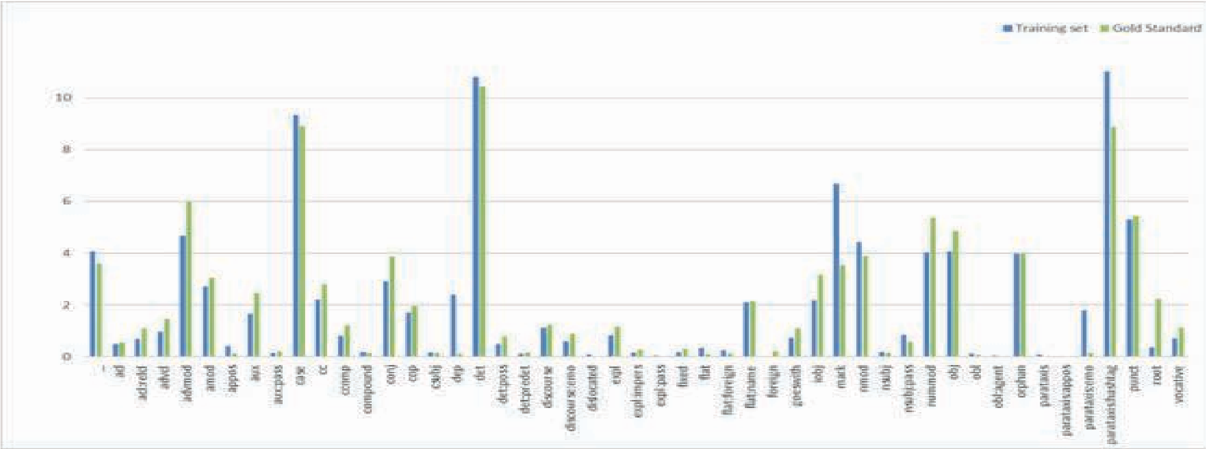


Figure 1. Frequency distribution of syntactic relation tags in the training set and the gold standard.

Figure 2. Frequency distribution of part-of-speech tags in the training set and the gold standard.

resulting scores on Universal part-of-speech tagging metric, XPOS on language-specific part-of-speech tagging metric, and UFeats on morphological features tagging metric. The last 5 rows show scores in five different parsing metrics.

What we found most challenging during the manual revision of the 1,000 sentences annotated automatically was correcting the errors in tokenization: many words that the tokenizer should have splitted were joined together. This type of tokenization error is often found when punctuation is used with non standard function. For example: we found that the token "oneste…volevo" ("honest…I wanted to") - an adjective, a punctuation mark and a verb - are conflated in a single token. In the manual revision, tokens like this have been splitted in three different tokens and other missing tags were added. The presence of such conflated words mayhave caused a worse score in the metric that evaluates the performance of segmentation, and consequentially in the other scores. The evaluation on the parser starts with aligning system nodes and gold nodes; their respective parent nodes are also considered; if the system parent is not aligned with the gold parent or if the relation label differs, the word is not counted as correctly attached. Despite errors in segmentation seem frequent in the corpus, this did not cause an excessive lowering of the scores on the various metrics reported in Table 2 and 3. Another error that appears frequently regards the lemma assigned to the abbreviations that are not present in PoSTTWITA-UD. Canonical abbreviations are tagged correctly, for example "cmq" for "comunque" ("however"). The abbreviations tagged incorrectly are those which appeared few times: such as "ql" that stands for "quelli" (those). An unexpected good result has been achieved on parsing metrics. This result could be due to the "preference of UD scheme in assigning headedness to content words" (Sanguinetti et al., 2018); therefore, the tendency of the social media languages to eliminate function words does not affect the performance of the parser. Another explanation can be found in the very similar frequencies distribution of part-of-speeches and syntactic relations in the training set and the gold standard, as shown in Figure 1 and 2.

Overall, the model trained on PoSTWITA-UD turned out to perform well on the test set of the ConteCorpus because PoSTWITA-UD tagset has been adapted with attention to some recurrent features of social media languages. Our evaluation showed that a model trained on texts retrieved by social media can adapt well to other social media texts if one pays attention to the neural architecture of the model and the annotation format being used.

5 Topic Modeling

To provide an overall view of the content of this large corpus we performed a Topic Modeling training and testing an LDA model on the ConteCorpus.

5.1 Methodology

Topic	1: Economics	2: Prime Minister	3: Politics	4: Pandemic	5: Home
Terms	pagare, soldo, italia, euro, chiudere, mese, debito, azienda, prestito, lavorare	presidente, grazie, Conte, lavoro, bravo, italia, italiano, signore, giuseppe, caro	italiano, europa, italia, paese, banca, popolo, governo, chiedere, germania, storia	uscire, miliardo, firmare, virus, decreto, Salvini, maria, pandemia, chiedere, italy	sperare, casa, aspettare, perdere, impresa, tedesco, subito, tempo, fondo, stipendio
English Translation	to pay, money, italy, euro, to close, month, loan, company, to work	prime minister, thank you, Conte, work, bravo, italy, italian, sir, giuseppe, dear	italian, europe, italy, country, bank, people, government, to ask, germany, story	to go out, billion, to sign, virus, decree, Salvini, maria, pandemic, to ask, italy	to hope, home, to wait, to lose, business, german, immediately, time, capital, salary

Table 4. Topic generated from the LDA model and the ten most frequent terms.

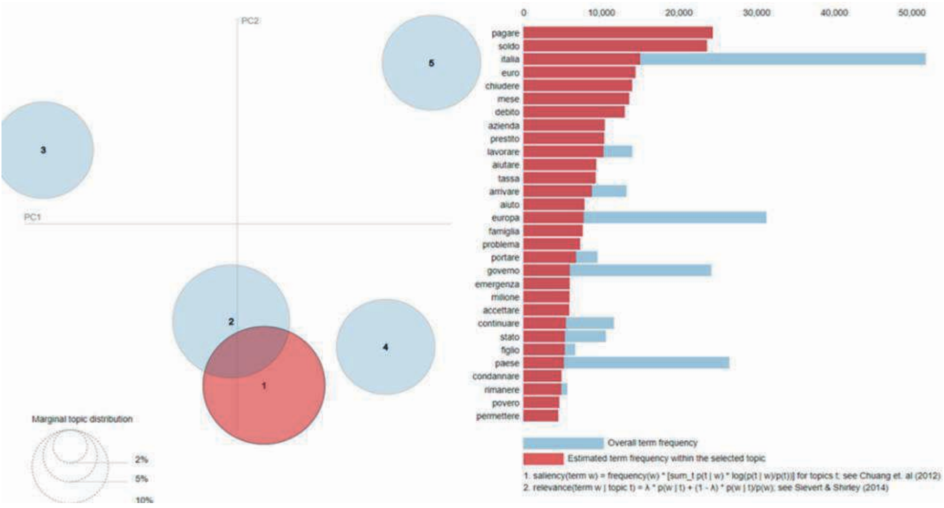


Figure 3. Intertopic distance Map and Top-30 most relevant terms for Topic 1. For a better view visit: https://sites.google.com/view/ldavisualizationcontecorpus/home-page.

To perform topic modelling, we sampled 10% of the sentences in our dataset and trained a LDA model. We treated each sentence as a document. We pre-processed lemmas removing stopwords, downloading Italian stopwords list from the NLTK (Natural Language Toolkit) library[10] and manually inserting missing stopwords. We filtered out tokens that appear in less than 15 documents and tokens with less than three letters; additionally, we kept only the 100,000 most frequent words. We transformed the documents into vectors creating a bag-of-words representation of each document. Then, we performed the term frequency-inverse document frequency (TF-IDF) on the whole corpus to assign higher weights to the most important words. Gensim LDA model[11] was applied first to the bags-of-words and secondly on the TF-IDF corpus to extract latent topics. Better performances were achieved with the LDA model applied to bags-of-words. We determined the optimal number of topics in LDA using the Coherence Value metric.[12] The underlying idea is that a good model will generate topics with high topic Coherence Value score. We ran different LDA experiments varying the number of topics and selected the model with the highest medium topic Coherence Value score. Our final model generated 5 topics and has a topic medium Coherence Value score of 0.5. Table 4 illustrates the top ten most representative terms associated with each detected topic.

5.2 Results

As expected, all the topics extracted from the corpus are related to the concerns about the emergency. The focus is on the economic aspect of the emergency. The first ten most frequent words in *Economics* topic (Table 4 and Figure 3) are economic terms: “loan”, “company”, “to pay” “money” etc. In all the other topics at least one of the 10 most frequent words comes from the economic sphere. Among the ten most frequent words of each topic there are only two words regarding the pandemic, found in *Pandemic* topic: "virus" and "pandemic". It is no coincidence that the most frequent word in this topic is “to go out”. The need to face the emergency through the intervention of the institutions is evident. This is shown espe

[10] https://www.nltk.org/.
[11] https://radimrehurek.com/gensim/models/ldamodel.html.

[12] Coherence Value metric is developed by Roder (2015). It evaluates a single topic by measuring the degree of semantic similarity between high scoring words in the topic.

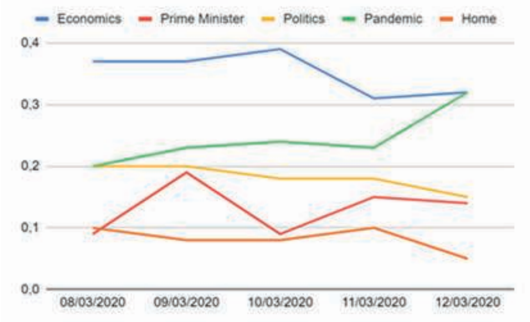


Figure 4. Prevalence of topics during the days 8-12 March 2020.

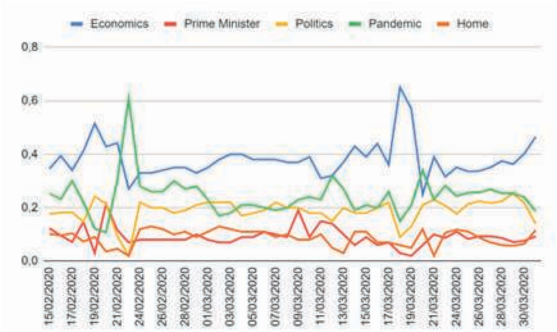


Figure 5. Prevalence of topics during the days 15 February-30 March 2020.

cially by *Prime Minister* and *Politics* topics (Table 4). *Prime Minister* topic most frequent words are related to the Prime Minister. Perhaps words like “bravo” and “thank you” and “dear” show a positive judgement towards him. In *Politics* topic one finds words of the institutional sphere such as: “country”, “government”, “people”, “bank”. *Home* topic is related to the private sphere with words like “to hope”, “home”, “to wait”, “to lose”, although there is no shortage of words from the economic sphere. In Figure 3 the distance between the centre of the circles indicates the similarity between the topics. Here you can see that only *Economics* topic and *Prime Minister* topic overlap; this indicates that the two topics are more similar with respect to the other topics. Moreover, the size of the area of each circle represents the importance of the topic relative to the corpus. *Economics* topic is the most important topic in the corpus. Finally, we tested our model on unseen documents: the comments published between 15 February and 30 March 2020, before and after the announcement of the first period of restrictions to combat the pandemic, and between 1 October and 14 November 2020, before and after the announcement of the second period of restrictions. Figures 4, 5 and 6 show trends in topics over time. Each line represents a topic and the x-axis shows the time progression. On 23 February, the first restrictive policies were announced for some Italian

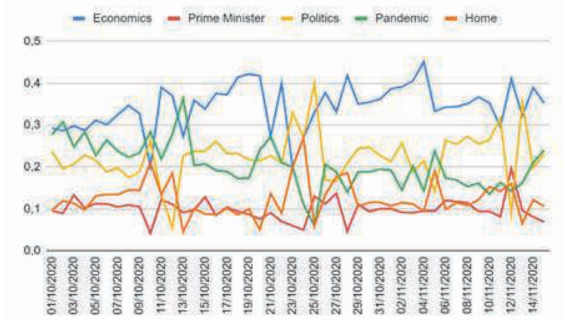


Figure 6. Prevalence of topics during the days 1 October-15 November 2020.

cities: Figure 5 shows a peak in the pandemic topic on that day. Figure 4 shows how the prevalence of the five topics changes on 8-12 March 2020. The Figure shows a peak on 9 March in *Prime Minister* topic: on that day he announced the first national restrictions period to combat the pandemic. Overall, the prevalent topics on those days are economics and pandemic. On 13 October, after a summer without major restrictions, with a new exponential increase in the curve of contagions, the Italian Parliament passed a decree limiting the possibility of aggregation. That day we have a new peak in the *Pandemic* theme (Figure 6). In the days that followed, the prevailing topic is *Economics*: on 28 October, the “ristoro” decree was approved to financially support commercial activities. A peak in the topic of *Economics* occurred on 18 March: on those days, discussions were taking place on whether to ask the European Union for financial aid to overcome the pandemic. The prevailing topics are therefore usually related to current events.

6 Concluding Observations

As mentioned before, models trained with data from social media are hardly generalizable. This stems from the fact that from a sociolinguistic perspective, the language of social media does not constitute a single variety. So, we expected that the results in the various evaluation metrics we performed would be worse than the results in the evaluation conducted on the PoSTWITA-UD test set. Surprisingly, in some metrics the results on evaluating the ConteCorpus test set were better than the results on the PoSTWITA-UD test set. To offer an overall view of the content of the ConteCorpus we performed topic modeling. The topics generated by the LDA model cover various aspects of the pandemic emergency, with a preponderance of political and economic issues. Unexpectedly, topics identified do not show concern regard the risk of contagion and the possibility of catching the disease.

References

Abd-Alrazaq, A., Alhuwail, D., Househ, M., Hamdi M., & Shah, Z. (2020). Top concerns of tweeters during the COVID-19 pandemic: infoveillance study. Journal of medical Internet research, 22(4), e19016.

Ahmed, M. E., Rabin, M. R. I., & Chowdhury, F. N. (2020). COVID-19: Social media sentiment analysis on reopening. arXiv preprint arXiv:2006.00804.

Amara, A., Taieb, M. A. H., & Aouicha, M. B. (2021). Multilingual topic modeling for tracking COVID-19 trends based on Facebook data analysis. Applied Intelligence, 51(5), 3052-3073.

Andreadis, S., Antzoulatos, G., Mavropoulos, T., Giannakeris, P., Tzionis, G., Pantelidis, N., ... & Kompatsiaris, I. (2021). A social media analytics platform visualising the spread of COVID-19 in Italy via exploitation of automatically geotagged tweets. Online Social Networks and Media, 23, 100134.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003) Latent dirichlet allocation, the Journal of machine Learning research (JMach), 3, 993–1022.

Chakraborty, K., Bhatia, S., Bhattacharyya, S., Platos, J., Bag, R., & Hassanien, A. E. (2020). Sentiment Analysis of COVID-19 tweets by Deep Learning Classifiers—A study to show how popularity is affecting accuracy in social media. Applied Soft Computing, 97, 106754.

Crystal, D. (2001). Language and the Internet. Cambridge University Press.

Dashtian, H. and Murthy, D. (2021). Cml-covid: A large-scale covid-19 twitter dataset with latent topics, sentiment and location information. arXiv preprint arXiv:2101.12202.

De Santis, E., Martino, A., & Rizzi, A. (2020). An Infoveillance System for Detecting and Tracking Relevant Topics from Italian Tweets During the COVID-19 Event. IEEE Access, 8, 132527-132538.

Fiorentino, G. (2013). "Wild language" goes Web: new writers and old problems in the elaboration of the written code. In E. Miola (Ed.), Languages Go Web. Standard and non-standard languages on the Internet (pp. 67-90.). Alessandria, Edizioni dell'Orso.

Dozat, T. and Manning, C. D. (2017). Deep biaffine attention for neural dependency parsing. In Proceedings of the 2017 International Conference on Learning Representations (ICLR).

Duong, V., Luo, J., Pham, P., Yang, T., & Wang, Y. (2020). The ivory tower lost: How college students respond differently than the general public to the covid-19 pandemic. IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (pp. 126-130).

Feng, Y. and Zhou, W. (2020). Is working from home the new norm? an observational study based on a large geo-tagged covid-19 twitter dataset. arXiv preprint arXiv:2006.08581.

Gagliardi, G., Gregori, L., & Suozzi, A. (2021). L'impatto emotivo della comunicazione istituzionale durante la pandemia di Covid-19: uno studio di Twitter Sentiment Analysis. Proceedings of the Seventh Italian Conference on Computational Linguistics, CLiC-it 2020, Bologna, Italy. Volume 2769 of CEUR Workshop Proceedings.

Garcia, K. and Berton, L. (2021). Topic detection and sentiment analysis in Twitter content related to COVID-19 from Brazil and the USA. Applied Soft Computing, 101, 107057.

Gozzi, N., Tizzani, M., Starnini, M., Ciulla, F., Paolotti, D., Panisson, A., & Perra, N. (2020). Collective Response to Media Coverage of the COVID-19 Pandemic on Reddit and Wikipedia: Mixed-Methods Analysis. Journal of medical Internet research, 22(10), e21597.

Gupta, V., Jain, N., Katariya, P., Kumar, A., Mohan, S., Ahmadian, A., & Ferrara, M. (2021). An emotion care model using multimodal textual analysis on COVID-19. Chaos, Solitons & Fractals, 144, 110708.

Hussain, A., Tahir, A., Hussain, Z., Sheikh, Z., Gogate, M., Dashtipour, K., et al. (2021). Artificial Intelligence–Enabled Analysis of Public Attitudes on Facebook and Twitter Toward COVID-19 Vaccines in the United Kingdom and the United States: Observational Study. Journal of medical Internet research, 23(4), e26627.

Jelodar, H., Wang, Y., Orji, R., & Huang, S. (2020). Deep sentiment classification and topic discovery on novel coronavirus or covid-19 online discussions: Nlp using lstm recurrent neural network approach. IEEE Journal of Biomedical and Health Informatics, 24(10), 2733-2742.

Kruspe, A., Häberle, M., Kuhn, I., & Zhu, X. X. (2020). Cross-language sentiment analysis of European Twitter messages during the COVID-19 pandemic. arXiv preprint arXiv:2008.12172.

Lamsal, R. (2020). Design and analysis of a large-scale COVID-19 tweets dataset. Applied Intelligence, 1-15.

Lomborg, S., & Bechmann, A. (2014). Using APIs for data collection on social media. The Information Society, 30(4), 256-265.

Low, D. M., Rumker, L., Talkar, T., Torous, J., Cecchi, G., & Ghosh, S. S. (2020) Natural Language Processing Reveals Vulnerable Mental Health Support Groups and Heightened Health Anxiety on Reddit During COVID-19: Observational Study. Journal of medical Internet research, 22(10), e22635.

Naseem, U., Razzak, I., Khushi, M., Eklund, P. W., & Kim, J. (2021). COVIDSenti: a large-scale benchmark Twitter data set for COVID-19 sentiment analysis. IEEE transactions on computational social systems.

Nemes, L. and Kiss, A. (2021). Social media sentiment analysis based on COVID-19. Journal of Information and Telecommunication, 5(1), 1-15.

Ordun, C., Purushotham, S., & Raff, E., (2020). Exploratory analysis of covid-19 tweets using topic modeling, umap, and digraphs. arXiv preprint arXiv:2005.03082.

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. Association for Computational Linguistics (ACL) System Demonstrations.

Röder, M., Both, A., & Hinneburg, A. (2015). Exploring the space of topic coherence measures. Proceedings of the eighth ACM international conference on Web search and data mining (pp. 399–408).

Sanguinetti, M., Bosco, C., Lavelli, A., Mazzei, A., Antonelli, O., & Tamburini, F. (2018, May). PoSTWITA-UD: an Italian Twitter Treebank in universal dependencies. Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018).

Sciandra, A., (2020). COVID-19 Outbreak through Tweeters' Words: Monitoring Italian Social Media Communication about COVID-19 with Text Mining and Word Embeddings. 2020 IEEE Symposium on Computers and Communications (ISCC) (pp. 1-6), IEEE.

Stella, M., Restocchi, V., & De Deyne, S., (2020). #lockdown: Network-enhanced emotional profiling in the time of Covid-19. Big Data and Cognitive Computing, 4(2), 14.

Stella, M., (2020). Cognitive network science reconstructs how experts, news outlets and social media perceived the COVID-19 pandemic. Systems, 8(4), 38.

Stella, M., Vitevitch, M. S., & Botta F., (2021) Cognitive networks identify the content of English and Italian popular posts about COVID-19 vaccines: Anticipation, logistics, conspiracy and loss of trust. arXiv preprint arXiv:2103.15909.

Su, Y., Xue, J., Liu, X., Wu, P., Chen, J., Chen, C., et al. (2020). Examining the impact of COVID-19 lockdown in Wuhan and Lombardy: a psycholinguistic analysis on Weibo and Twitter. International journal of environmental research and public health, 17(12), 4552.

Sullivan, K. J., Burden, M., Keniston, A., Banda, J. M., & Hunter, L. E. (2020). Characterization of Anonymous Physician Perspectives on COVID-19 Using Social Media Data. Pac Symp Biocomput.

Tamburini, F. (2020). EmoItaly. http://corpora.ficlit.unibo.it/EmoItaly/.

Trevisan, M., Vassio, L., & Giordano, D. (2021). Debate on online social networks at the time of COVID-19: An Italian case study. Online Social Networks and Media, 23, 100136.

Wang, J., Zhou, Y., Zhang, W., Evans, R., & Zhu, C. (2020). Concerns Expressed by Chinese Social Media Users During the COVID-19 Pandemic: Content Analysis of Sina Weibo Microblogging Data. Journal of medical Internet research, 22(11), e22152.

Vitale, P., Pelosi, S., Falco, M. (2020). #andràtuttobene: Images, Texts, Emojis and Geodata in a Sentiment Analysis Pipeline. Proceedings of the Seventh Italian Conference on Computational Linguistics, CLiC-it 2020, Bologna, Italy. Volume 2769 of CEUR Workshop Proceedings. http://ceur-ws.org/Vol-2769/paper_62.pdf.

PROTECT
A Pipeline for Propaganda Detection and Classification

Vorakit Vorakitphan, Elena Cabrio, Serena Villata
Université Côte d'Azur, Inria, CNRS, I3S, France
vorakit.vorakitphan@inria.fr,
elena.cabrio@univ-cotedazur.fr, villata@i3s.unice.fr

Abstract

English. Propaganda is a rhetorical technique to present opinions with the deliberate goal of influencing the opinions and the actions of other (groups of) individuals for predetermined misleading ends. The employment of such manipulation techniques in politics and news articles, as well as its subsequent spread on social networks, may lead to threatening consequences for the society and its more vulnerable members. In this paper, we present PROTECT (PROpaganda Text dEteCTion), a new system to automatically detect propagandist messages and classify them along with the propaganda techniques employed. PROTECT is designed as a full pipeline to firstly detect propaganda text snippets from the input text, and then classify the technique of propaganda, taking advantage of semantic and argumentation features. A video demo of the PROTECT system is also provided to show its main functionalities.

Italiano. *La propaganda è una tecnica retorica per presentare determinate opinioni con l'obiettivo deliberato di influenzare le opinioni e le azioni di altri (gruppi di) individui per fini predeterminati e tendenzialmente fuorvianti. L'impiego di tale tecnica di manipolazione in politica e nella stampa, così come la sua diffusione sulle reti sociali, può portare a conseguenze disastrose per la società e per i suoi membri più vulnerabili. In questo articolo presentiamo PROTECT (PROpaganda Text dEteCTion), un nuovo sistema per identificare automaticamente i messaggi propagandistici e classificarli rispetto alle tecniche di propaganda utilizzate. PROTECT è un sistema progettato come una pipeline completa per rilevare in primo luogo i frammenti di testo propagandistici dato il testo proposto, e successivamente classificare tali frammenti secondo la tecnica di propaganda usata, sfruttando le caratteristiche semantiche e argomentative del testo. Questo articolo presenta anche un video dimostrativo del sistema PROTECT per mostrare le principali funzionalità fornite all'utente.*

1 Introduction

Propaganda represents an effective but often misleading communication strategy which is employed to promote a certain viewpoint, for instance in the political context (Lasswell, 1938; Koppang, 2009; Dillard and Pfau, 2009; Longpre et al., 2019). The goal of this communication strategy is to persuade the audience about the goodness of such a viewpoint by means of misleading and/or partial arguments, which is particularly harmful for the more vulnerable public in the society (e.g., young or elder people). Therefore the ability to detect the occurrences of propaganda in political discourse and newspaper articles is of main importance, and Natural Language Processing methods and technologies play a main role in this context addressing the propaganda detection and classification task (Da San Martino et al., 2019; Da San Martino et al., 2020a). It is, in particular, important to make this vulnerable public aware of the problem and provide them tools able to raise their awareness and develop their critical thinking.

To achieve this ambitious goal, we present in this paper a new tool called PROTECT (PROpaganda Text dEteCTion) to automatically identify and classify propaganda in texts. In the current version, only English text is processed. This tool has been designed with an easy-to-access user interface and a web-service API to ensure a wide public use of PROTECT online. To the best of our knowledge, PROTECT is the first online tool for propagandist text identification and classification with an interface allowing the user to submit his/her own text to be analysed.[1]

PROTECT presents two main functionalities: *i)* the automatic propaganda detection and classification service, which allows the user to paste or upload a text and returns the text where the propagandist text snippets are highlighted in different colors depending on the propaganda technique which is employed, and *ii)* the propaganda word clouds, to show in a easy to catch visualisation the identified propagandist text snippets. PROTECT is deployed as a web-service API, allowing users to download the output (the text annotated with the identified propaganda technique) as a `json` file. The PROTECT tool relies on a pipeline architecture to first detect the propaganda text snippets, and second to classify the propaganda text snippets with respect to a specific propaganda technique. We cast this task as a sentence-span classification problem and we address it relying on a transformer architecture. Results reach SoTA systems performances on the tasks of propaganda detection and classification (for a comparison with SoTA algorithms, we refer to (Vorakitphan et al., 2021)).

The paper is structured as follows: first, Section 2 discusses the state of the art in propaganda detection and classification and compares our contribution to the literature. Then Section 3 describes the pipeline for the detection and classification of propaganda text snippets as well as the data sets used for the evaluation and the obtained results. Section 4 describes the functionalities of the web interface, followed by the Conclusions.

2 Related Work

In the last years, there has been an increasing interest in investigating methods for textual propaganda detection and classification. Among them, (Barrón-Cedeño et al., 2019) present a system to organize news events according to the level of propagandist content in the articles, and introduces a new corpus (QProp) annotated with the propaganda vs. trustworthy classes, providing information about the source of the news articles. Recently, a web demo named `Prta` (Da San Martino et al., 2020b) has been proposed, trained on disinformation articles. This demo allows a user to enter a plain text or a URL, but it does not allow users to download such results. Similarly to PROTECT, `Prta` shows the propagandist messages at the snippet level with an option to filter the propaganda techniques to be shown based on the confidence rate, and also analyzes the usage of propaganda technique on determined topics. The implementation of this system relies on the approach proposed in (Da San Martino et al., 2019).

The most recent approaches for propaganda detection are based on language models that mostly involve transformer-based architectures. The approach that performed best on the NLP4IF'19 sentence-level classification task relies on the BERT architecture with hyperparameters tuning without activation function (Mapes et al., 2019). (Yoosuf and Yang, 2019) focused first on the pre-processing steps to provide more information regarding the language model along with existing propaganda techniques, then they employ the BERT architecture casting the task as a sequence labeling problem. The systems that took part in the SemEval 2020 Challenge - Task 11 represent the most recent approaches to identify propaganda techniques based on given propagandist spans. The most interesting and successful approach (Jurkiewicz et al., 2020) proposes first to extend the training data from a free text corpus as a silver dataset, and second, an ensemble model that exploits both the gold and silver datasets during the training steps to achieve the highest scores.

As most of the above mentioned systems, also PROTECT relies on language model architectures for the detection and classification of propaganda messages, empowering them with a rich set of features we identified as pivotal in propagandist text from computational social science literature (Vorakitphan et al., 2021). In particular, (Morris, 2012) discusses how emotional markers and affect at word- or phrase-level are employed in propaganda text, whilst (Ahmad et al., 2019) show that the most effective technique to extract senti-

[1] The video demonstrating the PROTECT tool is available here `https://1drv.ms/u/s!Ao-qMrhQAfYtkzD69JPAYY3nSFub?e=oUQbxQ`

ment for the propaganda detection task is to rely on lexicon-based tailored dictionaries. (Li et al., 2017) show how to detect degrees of strength from calmness to exaggeration in press releases. Finally, (Troiano et al., 2018) focus on feature extraction of text exaggeration and show that main factors include imageability, unexpectedness, and the polarity of a sentence.

3 Propaganda Detection and Classification

PROTECT addresses the task of propaganda technique detection and classification at fragment-level, meaning that both the spans and the type of propaganda technique are identified and highlighted in the input sentences. In the following, we describe the datasets used to train and test PROTECT, and the approach implemented in the system to address the task.

3.1 Datasets

To evaluate the approach on which PROTECT relies, we use two standard benchmarks for Propaganda Detection and Classification, namely the NLP4IF'19 (Da San Martino et al., 2019) and SemEval'20 datasets (Da San Martino et al., 2020a). The former was made available for the shared task NLP4IF'19 on fine-grained propaganda detection. 18 propaganda techniques are annotated on 469 articles (293 in the training set, 75 in the development set, and 101 in the test set).[2] As a follow up, in 2020 SemEval proposed a shared task (T11)[3] reducing the number of propaganda categories with respect to NLP4IF'19 (14 categories, 371 articles in the training set and 75 in the development set). PROTECT detects and classifies the same list of 14 propaganda techniques as in the SemEval task, namely: *Appeal_to_Authority, Appeal_to_fear-prejudice, Bandwagon, Reductio_ad_hitlerum, Black-and-White_Fallacy, Causal_Oversimplification, Doubt, Exaggeration_Minimisation, Flag-Waving, Loaded-Language, Name-Calling_Labeling, Repetition, Slogans, Thought-terminating_Cliches, Whataboutism_Straw-Men_Red-Herring*.

Those classes are not uniformly distributed in the data sets. *Loaded-Language* and *Name-Calling_Labeling* are the classes with the higher number of instances (representing respectively 32% and 15% of the propagandist messages on all above-mentioned datasets). The classes with the lower number of instances are *Whataboutism, Red-Herring, Bandwagon, Straw-Men*, respectively occurring in 1%, 0.87%, 0.29%, 0.23% in NLP4IF'19 datasets. In SemEval'20T11 such labels where merged, and the classes *Whataboutism_Straw-Men_Red-Herring, Bandwagon* respectively represent 1.33% and 1.29% of the propagandist messages.

3.2 PROTECT Architecture

Given a textual document or a paragraph as input, the system performs two steps. First, it performs a binary classification at token level, to label a token as propagandist or not. Then, it classifies propagandist tokens according to the 14 propaganda categories from SemEval task (T11).

For instance, given the following example *"Manchin says Democrats acted like babies at the SOTU (video) Personal Liberty Poll Exercise your right to vote."* the snippets *"babies"* is first classified as propaganda (step 1), and then more specifically as an instance of the *Name-Calling_Labeling* propaganda technique (step 2).

Step 1: Propaganda Snippet Detection. To train PROTECT, we merge the training, development and test sets from NLP4IF, and the training set from Semeval'20 T11. The development set from Semeval'20 T11 is instead used to evaluate the system performances.[4] In the preprocessing phase, each sentence is tokenized and tagged with a label per token according to the IOB format.

For the binary classification, we adopt *Pretrained Language Model* (PLM) based on BERT (*bert-base-uncased* model) (Devlin et al., 2019) architecture. The hyperparameters are a learning rate of 5e-5, a batch of 8, max_len of 128. For the evaluation, we compute standard classification metrics[5] at the token-level. The results obtained by the binary classifier (macro average over 5 runs) on SemEval'20 T11 development set are 0.71 precision, 0.77 recall and 0.72 F-measure (us-

[2] https://propaganda.qcri.org/nlp4if-shared-task/

[3] https://propaganda.qcri.org/semeval2020-task11/

[4] The gold annotations of Semeval'20 test set are not available, this is why we selected the development set for evaluation.

[5] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html

Propaganda Technique	PLM: RoBERTa
Appeal_to_Authority	0.48
Appeal_to_fear-prejudice	0.57
Bandwagon,Reductio_ad_hit.	0.72
Black-White-Fallacy	0.38
Casual-Oversimplification	0.70
Doubt	0.74
Exaggeration,Minimisation	0.67
Flag-Waving	0.88
Loaded_Language	0.88
Name_Calling,Labeling	0.85
Repetition	0.70
Slogans	0.72
Thought-terminating_Cliches	0.52
Whatab.,Straw_Men,Red_Her.	0.55
Average	0.67

Table 1: Results on sentence-span classification on SemEval'20 T11 dev set (micro-F1) using span-pattern produced by the binary classification step (Step 1).

ing Softmax as activation function[6]).

We then perform a post-processing step to automatically join tokens labelled with the same propaganda technique into the same textual span.

Given that PLM is applied at token-level, each token is processed into sub-words (e.g., "running" is tokenized and cut into two tokens: "run" and "##ing"). Such sub-words can mislead the classifier. For instance, in the following sentence: "The next day, Biden said, he was informed by Indian press that there were at *least a few Bidens in* India.", our system detects *least a few Bidens in* as a propagandist snippet, but it misclassifies one sub-word ("at" was not considered as part of "at least", and therefore excluded from the propagandist snippet).

Step 2: Propaganda Technique Classification. We cast this task as a sentence-span multi-class classification problem. More specifically, both the tokenized sentence and the span are used to feed the transformer-based model RoBERTa (*roberta-base* pre-trained model)[7] (Liu et al., 2019) to perform both a sentence classification and a span classification. More precisely: *i)* we input a sentence to the tokenizer where max_length is set to 128 with padding; *ii)* we input the span provided by the propaganda span-template from SemEval T11 dataset, and we set max_length value of 20 with padding. RoBERTa tokenizer is applied in both cases. If a sentence does not contain propaganda spans, it is labeled as "none-propaganda".

To take into account context features at sentence-level, a BiLSTM is introduced. For each sentence, semantic and argumentation features are extracted following the methodology proposed in (Vorakitphan et al., 2021) and given in input to the BiLSTM model (hyper-parameters: 256 hidden_size, 1 hidden_layer, drop_out of 0.1 with ReLU function at the last layer before the joint loss function). Such features proved to be useful to improve the performances of our approach on propagandist messages classification, obtaining SoTA results on some categories (in (Vorakitphan et al., 2021) we provide a comparison of our model with SoTA systems on both NLP4IF and SemEval datasets).

To combine the results from sentence-span based RoBERTa with the feature-based BiLSTM we apply the joint loss strategy proposed in (Vorakitphan et al., 2021). Each model produces a loss per batch using CrossEntropy loss function L. Following the function: $loss_{\text{joint\_loss}} = \alpha \times \frac{(loss_{\text{sentence}} + loss_{\text{span}} + loss_{\text{semantic\_argumentation\_features}})}{N_{\text{loss}}}$ where each $loss$ value is produced from CrossEntropy function of its classifier (e.g., $loss_{\text{sentence}}$ and $loss_{\text{span}}$ from RoBERTa models of sentence and span, $loss_{\text{semantic\_argumentation\_features}}$ from the BiLSTM model.)

To train the above mentioned methods for the propaganda technique classification task, we merged the data sets of NLP4IF'19 and SemEval'20 T11 (same setting as in Step 1). Then we tested the full pipeline of PROTECT on the development set from Semeval'20 T11. The output of the snippet detection task (Step 1) are provided as a span-pattern to the models performing Step 2. Table 1 reports on the obtained results of the full pipeline (Step 1+Step 2) averaged over 5 runs (we cannot provide a fair comparison of those results with SoTA systems, given that in SemEval the two tasks are separately evaluated and no pipeline results are provided). We can notice however, that our results in a pipeline are comparable with the

[6]We are aware that sigmoid function is usually used as default activation function in binary classification. However, in our setting we tested both functions and we obtained better performances with Softmax as activation function (+0.04 F1 with respect to sigmoid).

[7]`https://huggingface.co/transformers/model_doc/roberta.html`

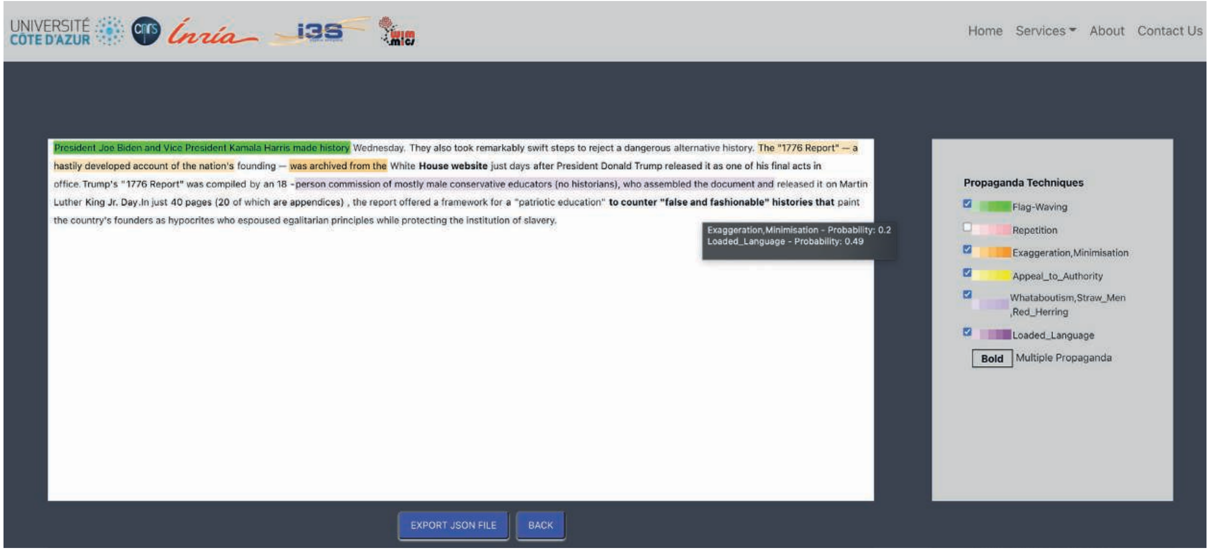


Figure 1: PROTECT Interface: Propaganda Techniques Classification

ones obtained in (Vorakitphan et al., 2021) on the two separate tasks.

Given the high complexity of the propaganda technique classification task and the classes' unbalance, some examples are miss-classified by the system. For instance, in the following sentence "The Mueller probe saw several within Trump's orbit indicted, but not *Trump , as family* or Trump himself", the system annotated the snippet in italics as "Name_Calling,Labeling", while the correct labels would have been "Repetition".

4 PROTECT Functionalities

As previously introduced, PROTECT allows a user to input plain text and retrieve the propagandist spans in the message as output by the system. In the current version of the system, two services are provided through the web interface (and the API), described in the following.

4.1 Service 1: Propaganda Techniques Classification

The system accepts an input plain text in English, and then the architecture described in Section 3.2 is run over such text. The output consists of an annotated version of the input text, where the different propagandist techniques detected by the system are highlighted in different colours. The colour of the highlighted snippet is distinctive of a certain propaganda technique: the darker the color, the higher the confidence score of the system in assigning the label to a textual snippet. Figure 1 shows an example of PROTECT web interface. Checkboxes on the right side of the page provide the key to interpret the colors, and allow the user to check or un-check (i.e. highlight or not) the different propagandist snippets in the text, filtering the results. Faded to dark colours represent the confidence level of the prediction (the darker the colour, the higher the system confidence). The snippets in bold contain multiple propaganda techniques in the same text spans, that can be unveiled hovering with the mouse over the snippets.

As said before, PROTECT can be used through the provided API, and annotated text can be downloaded as a JSON file with the detected propagandist snippet(s) at character indices (start to end indices of a snippet) based on individual sentence, propaganda technique(s) used, and the confidence score(s)).

4.2 Service 2: Propaganda Word Clouds

The propagandist snippets output by the system can also be displayed as word clouds, where the size of the words represents the system confidence score in assigning the labels (see Figure 2). The different sizes represent the confidence score of the prediction, and the colors the propaganda technique (as in Service 1). If multiple techniques are found in the same snippet, it is duplicated in the word cloud. As for the first service, a checkbox on the right side of the word clouds allows the user to select the propagandist techniques to be visualized. Also in this case, a json file can be downloaded with the system prediction.

The word cloud service has been added to PRO-

Figure 2: PROTECT Interface: Word Cloud

TECT in addition to the standard visualization, to provide a different and informative way to summarise propaganda techniques on a topic, and to facilitate their identification.

5 Conclusions

In this paper, we presented PROTECT, a propaganda detection and classification tool. PROTECT relies on a pipeline to detect propaganda snippets from plain text. We evaluated the proposed pipeline on standard benchmarks achieving state-of-the-art results. PROTECT is deployed as a web-service API that accepts a plain text input, returning downloadable annotated text for further usage. In addition, a propaganda word clouds service allows to gain further insights from such text.

Acknowledgments

This work is partially supported by the ANSWER project PIA FSN2 n. P159564-2661789/DOS0060094 between Inria and Qwant. This work has also been supported by the French government, through the 3IA Côte d'Azur Investments in the Future project managed by the National Research Agency (ANR) with the reference number ANR-19-P3IA-0002.

References

Siti Rohaidah Ahmad, Muhammad Zakwan Muhammad Rodzi, Nurlaila Syafira Shapiei, Nurhafizah Moziyana Mohd Yusop, and Suhaila Ismail. 2019. A review of feature selection and sentiment analysis technique in issues of propaganda. *International Journal of Advanced Computer Science and Applications*, 10(11).

Alberto Barrón-Cedeño, Israa Jaradat, Giovanni Martino, and Preslav Nakov. 2019. Proppy: Organizing the news based on their propagandistic content. *Information Processing & Management*, 56, 05.

Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeño, Rostislav Petrov, and Preslav Nakov. 2019. Fine-grained analysis of propaganda in news article. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5636–5646, Hong Kong, China, November. Association for Computational Linguistics.

Giovanni Da San Martino, Alberto Barrón-Cedeño, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. 2020a. SemEval-2020 task 11: Detection of propaganda techniques in news articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1377–1414, Barcelona (online), December. International Committee for Computational Linguistics.

Giovanni Da San Martino, Shaden Shaar, Yifan Zhang, Seunghak Yu, Alberto Barrón-Cedeño, and Preslav Nakov. 2020b. Prta: A system to support the analysis of propaganda techniques in the news. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 287–293, Online, July. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

James Price Dillard and Michael Pfau. 2009. *The Persuasion Handbook: Developments in Theory and Practice*. Sage Publications, Inc.

Dawid Jurkiewicz, Łukasz Borchmann, Izabela Kosmala, and Filip Graliński. 2020. ApplicaAI at SemEval-2020 task 11: On RoBERTa-CRF, span CLS and whether self-training helps them. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1415–1424, Barcelona (online), December. International Committee for Computational Linguistics.

Haavard Koppang. 2009. Social influence by manipulation: A definition and case of propaganda. *Middle East Critique*, 18:117 – 143.

Harold Dwight Lasswell. 1938. Propaganda technique in the world war.

Yingya Li, Jieke Zhang, and Bei Yu. 2017. An NLP analysis of exaggerated claims in science news. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 106–111, Copenhagen, Denmark, September. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692. _eprint: 1907.11692.

Liane Longpre, Esin Durmus, and Claire Cardie. 2019. Persuasion of the undecided: Language vs. the listener. In *Proceedings of the 6th Workshop on Argument Mining*, pages 167–176, Florence, Italy, August. Association for Computational Linguistics.

Norman Mapes, Anna White, Radhika Medury, and Sumeet Dua. 2019. Divisive language and propaganda detection using multi-head attention transformers with deep learning BERT-based language models for binary classification. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 103–106, Hong Kong, China, November. Association for Computational Linguistics.

Travis Morris. 2012. Extracting and networking emotions in extremist propaganda. In *2012 European Intelligence and Security Informatics Conference*, pages 53–59.

Enrica Troiano, Carlo Strapparava, Gözde Özbal, and Serra Sinem Tekiroğlu. 2018. A computational exploration of exaggeration. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3296–3304, Brussels, Belgium, October-November. Association for Computational Linguistics.

Vorakit Vorakitphan, Elena Cabrio, and Serena Villata. 2021. "Don't discuss": Investigating Semantic and Argumentative Features for Supervised Propagandist Message Detection and Classification. In *Recent Advances in Natural Language Processing (RANLP 2021)*, Varna (Online), Bulgaria, September.

Shehel Yoosuf and Yin Yang. 2019. Fine-grained propaganda detection with fine-tuned BERT. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 87–91, Hong Kong, China, November. Association for Computational Linguistics.

A Multi-Strategy Approach to Crossword Clue Answer Retrieval and Ranking

Andrea Zugarini[1,2], **Marco Ernandes**[1]
1. Expert.ai, Italy
2. DIISM University of Siena, Italy
{azugarini,mernandes}@expert.ai

Abstract

English. Crossword clues represent an extremely challenging form of Question Answering, due to their intentional ambiguity. Databases of previously answered clues are a vital source for the retrieval of candidate answers lists in Automatic Crossword Puzzles (CPs) resolution systems. In this paper, we exploit language neural representations for the retrieval and ranking of crossword clues and answers. We assess the performances of several embedding models, both static and contextual, on Italian and English CPs. Results indicate that embeddings usually outperform the baseline. Moreover, the use of embeddings for retrieval allows different ranking strategies, which turned out to be complementary, and lead to better results when used in combination.

Italiano. *Le domande dei cruciverba rappresentano una forma di Question Answering particolarmente complessa a causa della loro intenzionale ambiguità. I risolutori automatici di cruciverba sfruttano ampiamente basi di dati di domande precedentemente risposte. In questo articolo proponiamo l'uso di embeddings per la ricerca semantica di domande-risposte da tali databases. Le performances sono valutate in cruciverba di lingua sia italiana che inglese, confrontando diversi tipi di embeddings, sia contestuali che statici. I risultati suggeriscono che la ricerca semantica è migliore della baseline. Inoltre, l'utilizzo di embeddings permette di applicare differenti strategie di retrieval, che, migliorano la qualità dei risultati quando usate congiuntamente.*

1 Introduction

Crossword Puzzles (CPs) resolution is a popular game. As almost any other human game, it is possible to tackle the problem automatically. CPs solvers frame it into a constraint satisfaction task, where the goal is to maximize the probability of filling the grid with answers consistent with their clues and coherent to the puzzle scheme. These systems (Littman et al., 2002; Ernandes et al., 2005; Ginsberg, 2011) heavily rely on lists of candidate answers for each clue. Candidates' quality is crucial to CPs resolution. If the correct answer is not present in the candidates' list, the Crossword Puzzle cannot be solved correctly. Moreover, even a poorly ranked correct answer can lead to a failure in the crossword puzzle filling. Answers lists can come from multiple solvers, where each solver is typically specialized in solving different kinds of clues, and/or exploits different source of information. Such lists are mainly retrieved with two techniques: (1) by querying the web with search engines using clue representations; (2) interrogating clue-answer databases that contain previously answered clues. In this work, we focus on the latter.

In the problem of candidate answers retrieval from clue-answer knowledge sources, answers are ranked according to the similarity between a query clue and the clues in the DB. The similarity is provided by the search engine that assigns a score to each retrieved answer. Several approaches have been carried out to re-rank the candidates' list by means of learning to rank strategies (Barlacchi et al., 2014a; Barlacchi et al., 2014b; Nicosia et al., 2015; Nicosia and Moschitti, 2016; Severyn et al., 2015). These approaches require a training phase to learn how to rank and mostly differ for the re-ranking model or strategy adopted. In particular, pre-trained distributed representations and neural networks are used for re-ranking clues in (Severyn et al., 2015).

The re-ranking of answer candidates attempts to improve the quality of candidates' lists, assuming that the correct answer belongs to the list. Differently from previous work, we aim at directly retrieving richer lists of answer candidates from a clue-answer database. In order to do so, we exploit both static and contextual distributed representations to perform a semantic search on the DB. An embedding-based search extends the retrieval to semantically related clues that may be phrased differently. Moreover, it also allows us to map in the same space questions and answers, which opens the way for ranking answers directly based on their similarity with respect to the query clue. Our approach requires no training on CPs data and it can be applied with any pre-trained embedding model.

In summary, the contributions of this work are: (1) a semantic search approach to candidate answer retrieval in automatic crossword resolution; (2) two complementary retrieval methodologies (namely QC and QA) detecting candidate answers that when combined together (even naively) produce a better set of candidates; (3) a comparison between different pre-trained language representations (either static or contextual).

The paper is organized as follows. First, we describe in Section 2 distributed representations of language. In Section 3, we present the two answer retrieval approaches proposed in this work. Then, in Section 4 we outline the experiments in detail, and discuss the obtained results. Finally, we draw our conclusions in Section 5.

2 Language Representations

Assigning meaningful representations to language is a long standing problem. Since the inception of the first text mining solutions, the bag-of-words technique has been widely adopted as one of the standard approaches to text representation. Inverted indices and statistical weighting schemes (as TF-IDF or BM25) are still to this day commonly paired with bag-of-words, providing a scalable and effective approach to document retrieval. On the other hand, in the last decade, we have assisted to tremendous progress in the field of Natural Language Processing. Huge credit goes to the diffusion of distributed representations of words (Bengio et al., 2003; Mikolov et al., 2013a; Mikolov et al., 2013b; Collobert et al., 2011; Mikolov et al., 2018; Devlin et al., 2018) learned through Language Modeling related tasks on large corpora.

In general, the goal is to assign a fixed length representation of size d, aka embedding, to a textual passage s such that similar text passages - syntactically and/or semantically - are represented closely in such space. An embedding model f_e is a function mapping s to a d-dimensional vector, i.e: $f_e : s \to \mathbb{R}^d$. Since language is a composition of symbols (typically words), embedding models first tokenize text and then process such tokens in order to compute the representation of such textual passage.

Nowadays, there are lots of embedding models, and for some of them pre-trained embeddings are available in a plethora of languages (Yamada et al., 2020; Grave et al., 2018; Yang et al., 2019). Early methods like (Mikolov et al., 2013a) produce dense representations for single tokens - mainly words - therefore further processing is needed to obtain the actual representation of s, when s is composed of multiple words. These kinds of embeddings are also referred to as static embeddings, since the representation of a token is always the same regardless of the context in which it appears. In (Mikolov et al., 2018), authors extend (Mikolov et al., 2013a) introducing n-gram and sub-word information and in (Le and Mikolov, 2014), distributed representations are learned directly for sentences and documents.

Most of the proposed methods for contextual embeddings were based on recurrent neural language models (Melamud et al., 2016; Yang et al., 2019; Chidambaram et al., 2018; Mikolov et al., 2010; Marra et al., 2018; Peters et al., 2018), until the introduction of transformer architectures (Vaswani et al., 2017; Devlin et al., 2018; Liu et al., 2019) which are currently the state-of-the-art models. In the next Section we will discuss how such representations can be used to perform semantic search. In the experiments, we will exploit some of these embedding models - both static and contextual.

3 Semantic Search

Traditional CPs solvers rely on Similar Clue Retrieval mechanisms. The idea is to find possible

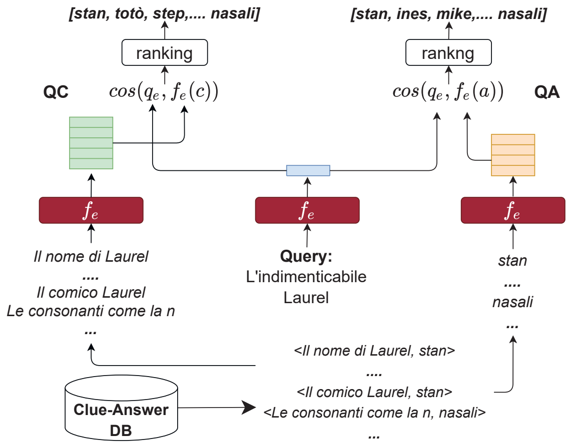


Figure 1: Sketch of the two answer candidates retrieval approaches: QC (on the left), QA (on the right). In QC, ranking is based on the similarity between the query embedding and all the clues in the DB, while in QA the similarity is computed between the query and the answers in the database.

answers from clues in the database that are similar to the given query. This is particularly effective for crosswords, since the same clues tend to be repeated over time, or may have little lexical variations. Retrieval of similar clues is based on search engines based on classical IR algorithms such as TF-IDF or BM25, representing clues in the database as documents to retrieve, given the target clue as query.

Here instead, we retrieve and rank documents with semantic search. We propose two strategies, namely QC and QA. QC is analogous to classical similar clues retrieval systems, with the difference that text is represented with a dense representation. The approach retrieves and ranks from the DB clues similar to the query and returns in output the answers associated to those clues. QA, instead, ranks the answers directly by computing the cosine distance between the query and the answers themselves. Intuitively, the latter approach ranks well answers semantically correlated to the question itself, particularly useful for clues about synonyms. As we will show in Section 4, due to their different nature, the list of candidates retrieved by the two approaches are strongly complementary. A sketch of the two approaches is outlined in Fig. 1. Let us describe them separately.

3.1 Similar Clues Retrieval

We are given a query clue which is a sequence of n words $q := (w_1, \ldots, w_n)$, and a clue-answer DB (C, A) constituted by M clue-answer pairs, where C and A indicate the list of all the clues and answers, respectively, while we denote a clue-answer pair as: (c, a).

We assign a fixed-length representation $q_e \in \mathbb{R}^d$ to the query clue q, computed with an embedding model:

$$q_e = f_e(q). \tag{1}$$

For contextual embeddings f_e is the model itself, since they work directly on the sequence, whereas for static embeddings we have to collapse n word representations together into a single vector. For simplicity, we simply average such embeddings.

Analogously, each clue $c \in C$ is encoded as in Equation 1. Then, we measure the cosine similarity between the query and each clue:

$$score(q, (c, a)) = cos(q_e, f_e(c)), \tag{2}$$

where $cos(\cdot, \cdot)$ denotes the cosine similarity. Thus, we obtain a similarity score for each clue-answer pair. In order to finally rank answers we average all clue-answer pairs having the same answer:

$$score(q, a) = \frac{1}{|\mathcal{A}|} \cdot \sum_{a_k \in \mathcal{A}} score(q, (c, a_k)), \tag{3}$$

where $\mathcal{A}$ indicates the set of clue-answer pairs where the answer a_k is equal to a. All the answers in A are then ranked. Since we know a priori the length of a query answer, candidates with incorrect lengths are filtered out. We refer to this approach as QC (Query-Clue).

3.2 Similar Answers Retrieval

Since we can map text into a fixed-length space, we can also rank by measuring the similarity between the query and the answer itself. The query is encoded exactly as in Equation 1. In this case however we only need the clue-answer DB to retrieve the set of unique answers, denoted as $\mathbf{A}$. Similarly to Equation 2, we compute the cosine similarity between query and answer embeddings:

$$score(q, a) = cos(q_e, f_e(a)), \tag{4}$$

for each $a \in \mathbf{A}$, then we rank as in QC. We call it QA (Query-Answer). It is important to remark that QA is only feasible using latent representations, traditional methods like TF-IDF are not suited because of their sparsity of representations. Moreover, QA is somewhat an orthogonal strategy with respect to QC. We will see in Section 4, how even a trivial ensemble of QA and QC is beneficial to the performances.

4 Experiments

In the experiments we aim to prove the effectiveness of semantic search to retrieve accurate lists of candidate answers, and to show that the QA approach carries out complementary information that can increase the coverage of the retrieval.

4.1 Experimental Setup

We considered for our experiments three well known embedding models, two static (Word2Vec[1 2], FastText[3]) and one contextual (Universal Sentence Encoder[4]), briefly denoted as W2V, FT and USE, respectively. We exploited pre-trained models for all of them. In absence of an Italian USE model, we used for the Italian crosswords database, the multilingual version of USE, that was trained on 16 languages (Italian included). Embedding models are compared against TF-IDF, which is a typical text representation in document retrieval problems.

To measure performances, we used well known metrics of Retrieval systems. In particular we considered Mean Hit at k (MH@k) and Mean Reciprocal Rank (MRR). Hit at k is 1 if the correct answer is within the first k elements of the list, 0 otherwise. The hits at k are evaluated for $k = \{1, 5, 20, 100\}$. MRR is defined as follows: $\frac{1}{n}\sum_{q=1}^{n}\frac{1}{rank(q)}$.

4.2 Datasets

We consider two different clue-answer databases for our experimentation. In particular, experiments were carried out on two languages, Italian and English, respectively on CWDB dataset (Barlacchi et al., 2014a) and New York Times Crosswords. We apply the same pre-processing pipeline in both corpora. (1) We discarded clue-answer pairs having answers with more than three characters, because they are typically about linguistic puzzles and they are addressed differently in CPs solvers. (2) Answer and clues containing special characters are erased. (3) Text has been lowercased and punctuation removed. (4) We kept only answers appearing in at least two clues.

[1]English: https://code.google.com/archive/p/word2vec/

[2]Italian: https://wikipedia2vec.github.io/wikipedia2vec/

[3]https://fasttext.cc/

[4]https://tfhub.dev/google/collections/universal-sentence-encoder/1

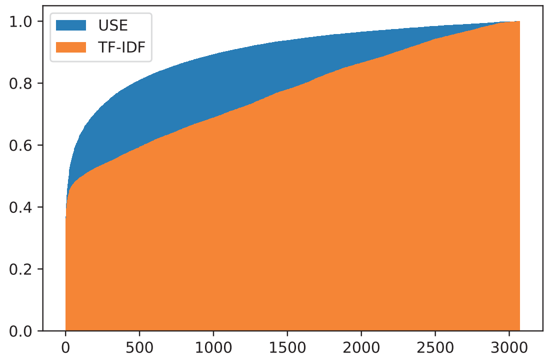


Figure 2: Comparison between cumulative density functions of ranking using USE (blue) and TF-IDF (orange) methods on English crosswords.

English Crosswords. The data consist of a collection of clue-answer pairs for crossword puzzles published in the New York Times[5] in 1997 and 2005, previously collected in (Ernandes et al., 2008). Overall, there are about $61,000$ clue-answer pair samples. Clues, answers and clue-answer pairs may occur multiple times. A clue is generally a short sentence, while answers are usually made up of a single word, but there are cases of multi-word answers. In such a case the answer is a string made of multiple words without any word separator. After pre-processing we obtain a corpus with $31,808$ pairs in which $27,527$ questions and $8,324$ answers are unique.

Italian Crosswords. The clue-answer database for Italian was constructed from *CWDB v0.1 it* corpus[6] (Barlacchi et al., 2014a). We combined pairs from both train and test splits, since we did not perform any training in our experiments and we opportunely omitted the clue-answer pair itself during its evaluation. From the original $62,011$ pairs, it remains $25,545$ pairs after pre-processing, constituted of $5,813$ unique answers and $16,970$ unique questions.

4.3 Results

All the results for Italian and English crosswords are outlined in Tables 1 and 2, respectively. From them, we can catch several interesting insights. First of all, contextual representations from Universal Sentence Encoders are gen-

[5]https://www.nytimes.com/

[6]https://ikernels-portal.disi.unitn.it/projects/webcrow

Model	Strategy	MH@1	MH@5	MH@20	MH@100	MRR
W2V	QA	**14.97**	**32.55**	**50.35**	**71.59**	**23.80**
FT	QA	6.78	14.47	26.88	52.46	11.44
USE	QA	7.89	17.81	29.30	46.80	13.24
TF-IDF	QC	**60.79**	66.43	68.53	72.62	63.54
W2V	QC	52.34	64.75	72.58	82.66	58.26
FT	QC	23.50	34.13	45.94	64.09	29.05
USE	QC	60.69	**70.93**	**76.81**	**84.70**	**65.57**
Ensemble$_{USE-W2V}$	QC-QA	-	**73.59**	**82.39**	**91.22**	-

Table 1: Evaluation of performances on CWDB Italian data. The best values of each column and strategy are marked in bold for both QC and QA methods.

Model	Strategy	MH@1	MH@5	MH@20	MH@100	MRR
W2V	QA	7.58	17.27	27.78	42.62	12.66
FT	QA	7.72	17.35	27.29	43.42	12.75
USE	QA	**8.63**	**19.69**	**30.01**	**45.17**	**14.25**
TF-IDF	QC	**26.15**	37.62	44.09	49.54	31.46
W2V	QC	19.63	31.69	42.66	57.38	25.65
FT	QC	15.72	24.32	32.67	46.64	20.20
USE	QC	25.78	**38.57**	**49.34**	**63.35**	**32.12**
Ensemble$_{USE-USE}$	QC-QA	-	**41.40**	**54.34**	**69.00**	-

Table 2: Evaluation of performances on English data. The best values of each column and strategy are marked in bold for both QC and QA methods.

erally the most effective ones, especially on similar clues retrieval (QC), where both the query and the elements to rank are textual sequences. Nonetheless, Word2vec embeddings work surprisingly well, outperforming FastText almost all the times. Furthermore, they are the best ones on QA search in Italian database. We believe the reason why Word2Vec outperforms USE on Italian QA is twofold. First, the advantage of contextual embeddings is less evident in QA setup, indeed USE brings less benefits on English QA as well. Second, USE is a multilingual model, therefore its embeddings are less specialized than Word2Vec which was instead trained for Italian only.

When comparing semantic search models against the baseline (TF-IDF) - which is only possible in QC - we can notice that, static embeddings struggle to outperform it. Indeed, the sparse nature of TF-IDF induces crisp similarity scores, very high for clues sharing the same keywords, extremely low for all the rest. On the contrary, similarity scores are more blurred with dense embeddings. As a consequence, TF-IDF achieves high MH@1 and MH@5 scores (and MRR too). However, TF-IDF leads to a poorer coverage when the candidates list grows (MH@20 and MH@100). This behavior is also evident in Fig. 2, where we compare the cumulative distributions of ranking with USE and TF-IDF. After the initial bump, TF-IDF hits growth is almost linear (i.e. random), whereas the Universal Sentence Encoder keeps growing significantly.

Ensembling QC and QA. Analyzing the results, we observed that ranks from QA and QC had low levels of overlaps. We reported in the last line of Tables 1 and 2, performances of a naive ensemble approach to combine QC and QA strategies. Due to the limited levels of overlaps, we decided to merge the two ranks taking the first $K/2$ ranks from each strategy to compute MH@K, $K = \{5, 20, 100\}$[7]. We chose the best embedding model on each strategy. Despite its simplicity and the large room for improvements, the ensemble significantly improved the performances in both languages. This suggests possible directions for further improving the retrieval of CPs solvers.

[7] Since K=5 is not even, we took the first 3 ranks from QC and the first two ranks from QA.

5 Conclusions

In this paper, we proposed two different semantic search strategies (QC and QA) for ranking and retrieving answer candidates to CPs clues. We exploited pre-trained state-of-the-art embeddings, both static and contextual, to rank clue-answer pairs from databases. Embedding-based retrieval overcomes some of the limitations of inverted indices models, leading to higher coverage ranks, and allowing similar answers retrieval (QA). Finally, we observed that, even a simple ensembling that combines QC and QA, is effective and improves overall retrieval performances.

This opens further research directions, where learning to rank methods could be exploited in order to better combine candidate answer lists from complementary approaches like QC and QA.

Acknowledgments

We thank Nicola Landolfi and Marco Maggini for the great support and fruitful discussions.

References

Gianni Barlacchi, Massimo Nicosia, and Alessandro Moschitti. 2014a. Learning to rank answer candidates for automatic resolution of crossword puzzles. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 39–48.

Gianni Barlacchi, Massimo Nicosia, and Alessandro Moschitti. 2014b. A retrieval model for automatic resolution of crossword puzzles in italian language. In *The First Italian Conference on Computational Linguistics CLiC-it 2014*, page 33.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Muthuraman Chidambaram, Yinfei Yang, Daniel Cer, Steve Yuan, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Learning cross-lingual sentence representations via a multi-task dual-encoder model. *arXiv preprint arXiv:1810.12836*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Marco Ernandes, Giovanni Angelini, and Marco Gori. 2005. Webcrow: A web-based system for crossword solving. In *AAAI*, pages 1412–1417.

Marco Ernandes, Giovanni Angelini, and Marco Gori. 2008. A web-based agent challenges human experts on crosswords. *AI Magazine*, 29(1):77–77.

Matthew L Ginsberg. 2011. Dr. fill: Crosswords and an implemented solver for singly weighted csps. *Journal of Artificial Intelligence Research*, 42:851–886.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR.

Michael L Littman, Greg A Keim, and Noam Shazeer. 2002. A probabilistic approach to solving crossword puzzles. *Artificial Intelligence*, 134(1-2):23–55.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Giuseppe Marra, Andrea Zugarini, Stefano Melacci, and Marco Maggini. 2018. An unsupervised character-aware neural approach to word and context representation learning. In *International Conference on Artificial Neural Networks*, pages 126–136. Springer.

Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional lstm. In *Proceedings of the 20th SIGNLL conference on computational natural language learning*, pages 51–61.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Massimo Nicosia and Alessandro Moschitti. 2016. Crossword puzzle resolution in italian using distributional models for clue similarity. In *IIR*.

Massimo Nicosia, Gianni Barlacchi, and Alessandro Moschitti. 2015. Learning to rank aggregated answers for crossword puzzles. In *European Conference on Information Retrieval*, pages 556–561. Springer.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Aliaksei Severyn, Massimo Nicosia, Gianni Barlacchi, and Alessandro Moschitti. 2015. Distributional neural networks for automatic resolution of crossword puzzles. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 199–204.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Ikuya Yamada, Akari Asai, Jin Sakuma, Hiroyuki Shindo, Hideaki Takeda, Yoshiyasu Takefuji, and Yuji Matsumoto. 2020. Wikipedia2Vec: An efficient toolkit for learning and visualizing the embeddings of words and entities from Wikipedia. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 23–30. Association for Computational Linguistics.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, et al. 2019. Multilingual universal sentence encoder for semantic retrieval. *arXiv preprint arXiv:1907.04307*.

Contributed Papers: Short Papers

Emerging Trends in Gender-Specific Occupational Titles in Italian Newspapers

Pierluigi Cassotti[1], **Andrea Iovine**[1], **Pierpaolo Basile**[1],
Marco De Gemmis[1] **and Giovanni Semeraro**[1]
1. Department of Computer Science, University of Bari Aldo Moro, Italy
{firstname}.{surname}@uniba.it

Abstract

The grammatical gender system can influence the way the semantic gender is perceived. Italian is a grammatical gender language, in which nouns are classified for gender. In this work, we investigate the usage of gender-specific forms of occupational titles in a diachronic corpus of 3 billion tokens extracted from two Italian newspapers. The hypothesis is that the usage of gender-specific forms might be influenced by socio-cultural aspects, such as changes in the employment policy. We automatically collect a set of occupational titles and perform a diachronic analysis exploiting the frequency of gender-specific forms. Results show a correlation between changes in the usage of gender-specific forms and socio-cultural events.

1 Introduction

Throughout history, the prerogative use of specific gender forms over particular professions can fade away by introducing changes in the language lexicon (e.g., neologisms) or in the language usage (e.g., word frequencies). The way the lexicon is affected by those changes depends on the grammatical gender system, i.e. the set of rules that define the agreement between noun classes forms and the other parts-of-speech. Grammatical gender systems can vary dramatically from one language to another. Gygax et al. (2019) propose a classification of languages based on their grammatical gender system. In this work, we focus on the Italian language, a grammatical gender language in which all nouns must be classified for gender. The Italian gender system admits three categories for nouns: gender-specific ending nouns, mobile gender nouns, and nouns where the gender is specified through determiners and adjectives (Marcato and Thüne, 2002). In gender-specific ending nouns, the gender forms are expressed through completely different lexical roots (e.g., *genero/nuora*). In mobile gender nouns, the specific gender forms share the same lexical root, and the semantic gender is instead represented by different suffixes (e.g., *scrittore/scrittrice*). In other cases, the semantic gender of a noun is inferred only by the determiner and/or adjective (e.g., *il* giudice, *la* giudice). The peculiar characteristic found in the Italian language has strong repercussions in the way people refer to occupational titles, because a specific gender form might be preferred over the other due to historical reasons, regardless of the gender of the actual person being talked about (Sabatini, 1985). This has become a hot-button issue in the last years, especially as a result of the United Nations Resolution “Transforming our world: the 2030 Agenda for Sustainable Development” with its global indicator framework for Sustainable Development Goals (SDGs), and specifically of SDG 5 *Achieve gender equality and empower all women and girls* (subgoal 5.1 *End all forms of discrimination against all women and girls everywhere*) (Lee et al., 2016).

The objective of this paper is to monitor how the use of gender-specific occupational titles has changed in the Italian language over the years through the use of diachronic analysis tools. We would like to emphasize that the goal is not to map the composition of men and women for each profession over time, as this cannot be reliably inferred from text. Instead, we are interested in gauging the cultural relevance of the gender-specific titles over time, as reflected in the news domain. Accordingly, the contributions in this paper can be summarized as follows:

(i) We analyze emerging trends in the use of gender-specific occupational titles in the Italian language in a corpus of newspaper articles.[1]

(ii) We perform a deep-dive analysis of the figures that have guided a significant shift for two professions in particular.

Large diachronic corpora have already been used to study social and cultural phenomenons that affected language in a significant way. The Google Ngrams Dataset (Goldberg and Orwant, 2013) is a dataset of n-grams extracted by 3.5 million books published between 1520 and 2008. Aiden and Michel (2011) exploit the huge quantity of information contained in the Google Ngrams Dataset to analyze the evolution of the language lexicon over time. In particular, the work offers interesting culturomics results, such as highlighting the spread of the term *influenza* during historical pandemic periods. Kutuzov et al. (2017) exploit diachronic word embeddings to track wars and conflicts that took place from 1994 to 2010 all around the world. Diachronic word embeddings are trained on the English Gigaword news corpus (Parker et al., 2011) and used to predict conflict states: *peace*, *war* and *stable*. Laine and Watson (2014) analyze the linguistic sexism occurring in *The Times* newspaper over five decades (1965-2005), relying on the classification of linguistic sexism proposed in (King, 1991). The authors hypothesize that occupational titles and agents would be more resistant to change than other forms of sexism over the decades. They confirm their hypothesis by exploring the frequencies of male and female affixes, showing that they keep stable. Burr (1995) performs an empirical analysis on manually-annotated occurrences of grammatical agents in a small synchronic corpus of Italian newspapers. The outcomes of this work lead the authors to conclude that women are underrepresented in Italian newspapers, especially in more high-position roles.

2 Corpus

Occupational titles occurrences are extracted from a diachronic corpus that comprises two sub-corpora. The former corpus is the "L'Unità" corpus (Basile et al., 2020) that covers the time period 1945-2014. The latter is crawled by the publicly available digital archive of the Italian newspaper "La Stampa" covering the period 1945-2005 and processed using the same methodology mentioned in (Basile et al., 2020). In order to align the two sub-corpora time ranges, we consider a sub-portion of the "L'Unità" corpus that spans the period 1948-2005. The overall corpus contains 3,529,820,155 tokens and spans the period 1948-2005. Corpus statistics are reported in Table 1. The corpus presents two main critical issues. First, despite having performed pre-processing and filtering, the documents from the earlier periods suffer from several OCR errors and noise. Second, data is not equally distributed, the number of tokens drops dramatically in the first years. Text is processed using the UDPipe model (Straka et al., 2016) included in spaCy[2]. The UDPipe model is trained on the Italian Stanford Dependency Treebank (Bosco et al., 2014). Each sentence is tokenized, lemmatized and annotated with PoS-tags, named entity tags and dependency relations. Moreover, the UDPipe model provides information about inflectional features of nouns exploited in the occupational titles extraction pipeline.

Corpus	Tokens	Period
L'Unità	425,833,098	1948-2014
La Stampa	3,145,959,127	1948-2005
Overall	3,529,820,155	1948-2005

Table 1: Corpus statistics.

3 Extracting Occupational Titles

The first step of our investigation consists of extracting a list of occupational titles from a common Knowledge Base. Specifically, we have exploited Wikidata (Vrandečić and Krötzsch, 2014), since it has collected a wide range of entities related to professional activities. We first extracted a list of all entities that are an instance of *profession* (wd:Q28640), or of an entity that is a subclass of it, for which a label in the Italian language is present. This label commonly contains the male gender form of the occupational title. Then, we filtered the list of professions by only including those that possess the *female form of label* (wdt:P2521) property for the Italian language. This property denotes the female variant of the occupational title, where applicable. The next step consists of filtering out occupational titles for which the gender is not easily distinguishable from text, such as those in which both gender variants

[1] All data collected in this experiment is available here: https://github.com/pierluigic/igsot

[2] https://spacy.io/

share the same lexical root (e.g. the aforementioned *il giudice/la giudice*), or those that do not feature gender variants at all (e.g. *la guardia*, i.e. the guard). We also removed all occupational titles that consist of two or more tokens. Then, we reduced the list by filtering out polysemous words. A common example of polysemy in the Italian language occurs when an occupational title shares the same lexical form as the discipline to which it belongs, such as *matematica* (female form of *mathematician*), or *fisica* (female form of *physicist*). For each occupational title, we used WordNet to find all synsets in which it appears and then removed it if the synset is a hyponym of the *discipline.n.01* synset. Moreover, we manually analyzed the list of remaining occupational titles and removed other instances of polysemy, which would otherwise hinder the quality of the results. For instance, we filtered the word *editrice* (female form of *editor*) as it can also appear in the phrase *casa editrice* (i.e. publishing house), and the word *tecnica* (female form of *technician*), which can also refer to the word *technique* depending on context. We also decided to remove words that have additional figurative meanings, such as *cacciatrice* (female form of *hunter*) and guerriera (female form of *warrior*). This process was undertaken by two independent annotators and then checked for agreement. The final result of this process is T, a set of tokens that unequivocally refer to occupational titles, and that feature distinct male and female gender variants which can reliably be extracted from text.

4 Experimental Setup

Once we have acquired the set of occupational titles T, the next step of the analysis consisted of measuring the frequency with which each term $w \in T$ occurs for each year in the corpus described in Section 2. We also make use of the lexical information contained in said corpus in order to eliminate any remaining ambiguity in the words. In fact, for each occupational title, we counted a hit in the corpus if it appears with the NOUN tag. This allows us to avoid counting occupational titles that can be confused with verbs or adjectives, such as *impiegato/impiegata*, which can refer to the noun *employee* in Italian, but also to the past participle conjugation of the verb *to employ*.

Moreover, we only counted a hit if the word has been registered with the singular form. This is done for two reasons: first, occurrences of the plural form are outside the scope of this investigation, because in Italian the male plural form is traditionally used as the default, while the female variant of the plural is only used in exceptional cases, such as when referring to a group that is composed entirely of women. Second, this strategy filters out cases where the plural form shares the same lexical root as one of the gender variants. An example of this is the word *infermiere* (i.e. *nurse*), which can refer to both the singular masculine form (as in *l'infermiere*), or the plural feminine form (as in *le infermiere*).

Since the objective of this study is to observe the trends in the use of masculine and feminine forms for occupational titles, we are interested in analyzing how their frequency changes from one year to the other. However, measuring the absolute frequency in each year for both forms would be misleading, as it heavily depends on the amount of data that is available for each year in the corpus. Instead, we compute the smoothed relative frequency p_w^t for each word w and each year t using the following formula:

$$p_w^t = \frac{f_w^t + 1}{C^t + \mid V^t \mid} \tag{1}$$

where f_w^t is the frequency of word w in the year t, C^t is the count of tokens occurring in the corpus the year t and $|V^t|$ is the vocabulary length computed on the year t. We compute p_w^t for both gender forms of each occupational title. Then we compute $odds(w)^t$ which represents the log ratio of the smoothed relative frequency of the female and male forms respectively:

$$odds(w)^t = log\frac{p_{w_f}^t}{p_{w_m}^t} \tag{2}$$

Operationally, $odds(w)^t$ specifies the probability that the feminine variant will appear in a text relative to the masculine form in the specified year t. We then obtain the time-series by concatenating the $odds(w)$ values computed for each year: $(odds(w)^{1948}, odds(w)^{1949}, .., odds(w)^{2004})$. Assuming a linear course of the time-series, three different scenarios can occur: *(i)* the occurrences of the female form are growing; *(ii)* the occurrences of the male form are growing; *(iii)* the ratio of the male and female form of an occupational title are stable over time. We com-

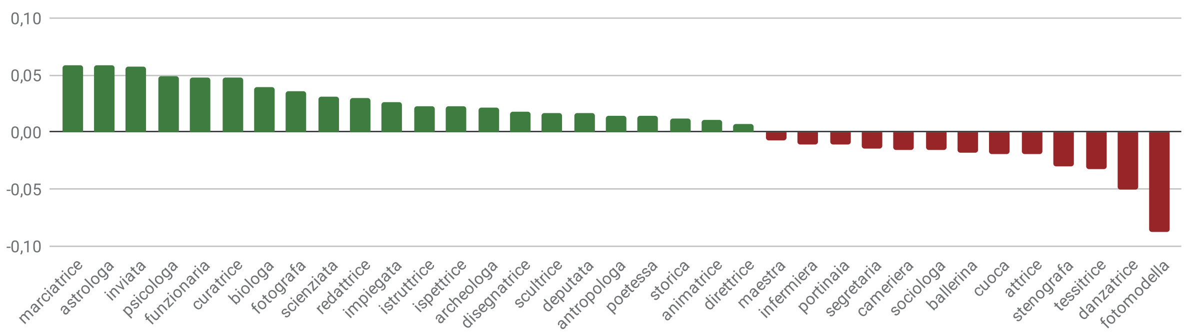


Figure 1: Final set of occupational titles (the female form is reported) and the slope of $odds(w)^t$.

puted the regression line of the time-series, using the linear least-squares regression method provided by the SciPy library[3]. We use the slope of the regression line to determine whether the values of $odds(w)^t$ are changing over time. If the slope is positive/negative, $odds(w)^t$ is increasing/decreasing over time, which means that the frequency of w_f is increasing/decreasing faster than that of w_m, or that the frequency of w_m is decreasing/increasing faster than that of w_f. For each regression line, we also compute the statistical significance of the slope parameter relying on the Wald Test (Fahrmeir et al., 2007). Specifically, the null hypothesis states that the slope parameter of the regression line is zero. In this stage, occupational titles for which we get a $p-value > 0.1$ are filtered out.

5 Results

Figure 1 describes the value of the slope for each occupational title. Depending on the sign of the slope, we can identify two distinct groups of occupational titles. Green bars indicate that the slope of $odds(w)^t$ is positive, i.e. the frequency of the feminine form is increasing relative to that of the masculine form. On the other hand, red bars indicate that the slope is negative, thus the frequency of the feminine form is decreasing relative to that of the masculine form. Out of 35 occupational titles, 22 have a positive slope, while 11 result in a negative slope. In particular, the most positive slope is the one associated to *marciat-ore/-rice* (i.e. *racewalker*), while the most negative slope is *fotomodell-o/-a* (i.e. *fashion model*).

For many of these titles, the resulting slope can be mapped to specific social changes. An interesting example in this regard is *infermiere* (i.e. *nurse*), to which a negative slope is recorded: indeed, in Italy the position of nurse has been opened to men starting from 1971[4]. The odds(w) time series of infermiera/infermiere is reported in Figure 2.

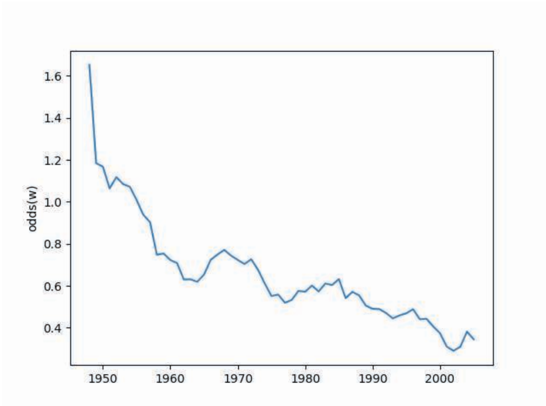


Figure 2: 10-year moving average of odds(w) for infermiera/infermiere.

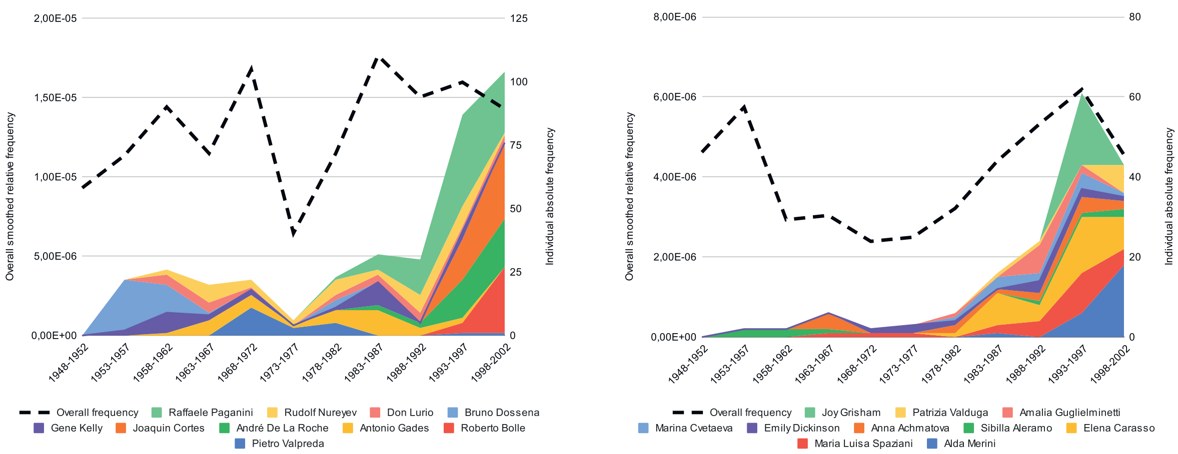
Moreover, results show that managerial roles such as *funzionaria* (i.e. *civil servant*), *ispettrice* (i.e. *inspector*), *direttrice* (i.e. *director*) are associated to a positive slope, which is indicative of a stronger perception of women in such roles.

A similar push can be observed also in the scientific domain, with a positive trend for the words *biologa* (i.e. *biologist*), *scienziata* (i.e. *scientist*), as well as the artistic one. On the other hand, we observe an increase in the usage of the masculine form for *segretario* (i.e. *secretary*), *ballerino* (i.e. *dancer*), and *stenografo* (i.e. *stenographer*).

In the second part of the experiment, we attempt to identify the people that have driven the change in the usage of the feminine and masculine forms of an occupational title. To do this, we retrieve

[3]https://www.scipy.org/

[4]https://www.gazzettaufficiale.it/eli/id/1971/04/03/071U0124/sg



(a) ballerino.

(b) poetessa.

Figure 3: Occurrences of Named Entities associated to two occupational titles. The X-axis reports the time periods. The left Y-axis reports the overall smoothed relative frequency of the occupational title. The right Y-axis reports the absolute frequency of each Named Entity.

the Named Entities (NEs) to which the occupational titles refer for each year, and monitor their frequency. In particular, we exploit the UDPipe annotations to extract valid NEs, i.e. entities that are directly connected to an occupational title via a dependency relation.

In Figure 3, we report the NEs extracted for two particular occupational titles: *ballerino* (i.e. male dancer) and *poetessa* (i.e. female poet). We have chosen these titles because they feature the largest number of occurrences of NEs in the corpus. The data is presented in the form of stacked line charts, which report the absolute frequency of each NE so that the height of a coloured line represents how many times a NE has been mentioned within a specified period. The dotted black line reports the overall smoothed relative frequency for the occupational title. Both the absolute frequency of NEs mentions and the overall smoothed relative frequency are aggregated in bins of 5 years.

Three male dancers are referenced over a wide period due to their historical role in the field: *Rudolf Nureyev*, *Antonio Gades* and *Gene Kelly*. However, the last years have seen a rise in popularity of new figures such as *Raffaele Paganini*, *Joaquin Cortes*, *André de La Roche* and *Roberto Bolle*.

Occurrences of specific female poets in the corpus keep low until the late ’70s. Ignoring a spike in 1953-1957, probably due to the quality issues in the data collected, the individual absolute frequency of NE mentions seems to agree with the overall smoothed relative frequency of the noun *poetessa*. In the 1988-2002 period, four figures overwhelm the scene: *Joy Grisham*, *Elena Carasso*, *Maria Luisa Spaziani* and *Alda Merini*. Even though the first work of *Maria Luisa Spaziani* dates back to 1954, we observe a significant rise in the occurrences in the early ’90s, when she is nominated three times for the Nobel Prize for Literature [5]. The increase in NE mentions over time is even more apparent in this case, however, it follows a different trend compared to that of the overall frequency of the noun *poetessa*, which suggests that the word may have been used differently in the earliest period.

6 Conclusion

This paper investigates the usage of gender-specific forms of occupational titles in the Italian language in a diachronic corpus of 3 billion tokens extracted from two popular Italian newspapers. Through this analysis, we show that there are significant changes in the way newspaper articles refer to the masculine and feminine form of an occupational title and that they are consistent with socio-cultural events, such as changes in the employment policy. Moreover, we performed a more fine-grained analysis by extracting the most influential figures that have guided this shift for two occupational titles (male dancers and female poets).

[5] https://en.wikipedia.org/wiki/Maria_Luisa_Spaziani

As future work, we propose to continue work on this field by increasing the size of the corpus and by including sources other than news, such as social media, job applications, and legal documents. This can help reduce any form of linguistic bias that may have been introduced by journalists and increase the significance of the results. Moreover, we will extend the list of occupational titles, as well as group titles together based on category. Finally, we propose to improve the process used to extract named entities that are associated with occupational titles in text.

Acknowledgments

This research has been partially funded by ADISU Puglia under the post-graduate programme "Emotional city: a location-aware sentiment analysis platform for mining citizen opinions and monitoring the perception of quality of life".

References

Erez Lieberman Aiden and Jean-Baptiste Michel. 2011. Culturomics: Quantitative Analysis of Culture Using Millions of Digitized Books. In *6th Annual International Conference of the Alliance of Digital Humanities Organizations, DH*, page 8, Stanford, CA, USA, June. Stanford University Library.

Pierpaolo Basile, Annalina Caputo, Tommaso Caselli, Pierluigi Cassotti, and Rossella Varvara. 2020. A Diachronic Italian Corpus based on "L'Unità". In Johanna Monti, Felice Dell'Orletta, and Fabio Tamburini, editors, *Proceedings of the Seventh Italian Conference on Computational Linguistics, CLiC-it 2020*, volume 2769 of *CEUR Workshop Proceedings*, Bologna, Italy, 3. CEUR-WS.org.

Cristina Bosco, Felice Dell'Orletta, Simonetta Montemagni, Manuela Sanguinetti, and Maria Simi. 2014. The EVALITA 2014 dependency parsing task. *The Evalita 2014 Dependency Parsing task*, pages 1–8.

Elisabeth Burr. 1995. Agentivi e sessi in un corpus di giornali italiani. In Gianna Marcato, editor, *Atti del Convegno Internazionale di studi Dialettologia al femminile*, pages 349–365, Padova, Italy, April. Cleup.

Ludwig Fahrmeir, Thomas Kneib, Stefan Lang, and Brian Marx. 2007. *Regression*. Springer.

Yoav Goldberg and Jon Orwant. 2013. A Dataset of Syntactic-Ngrams over Time from a Very Large Corpus of English Books. *Atlanta, Georgia, USA*, page 241.

Pascal Mark Gygax, Daniel Elmiger, Sandrine Zufferey, Alan Garnham, Sabine Sczesny, Lisa von Stockhausen, Friederike Braun, and Jane Oakhill. 2019. A Language Index of Grammatical Gender Dimensions to Study the Impact of Grammatical Gender on the Way We Perceive Women and Men. *Frontiers in Psychology*, 10:1604.

Ruth Elizabeth King. 1991. *Talking gender: A guide to nonsexist communication*. Copp Clark Professional.

Andrey Kutuzov, Erik Velldal, and Lilja Øvrelid. 2017. Tracing armed conflicts with diachronic word embedding models. In Tommaso Caselli, Ben Miller, Marieke van Erp, Piek Vossen, Martha Palmer, Eduard H. Hovy, Teruko Mitamura, and David Caswell, editors, *Proceedings of the Events and Stories in the News Workshop@ACL 2017*, pages 31–36, Vancouver, Canada, August. Association for Computational Linguistics.

Tarutuulia Laine and Greg Watson. 2014. Linguistic sexism in The Times-A diachronic study. *International Journal of English Linguistics*, 4(3):1.

Bandy X Lee, Finn Kjaerulf, Shannon Turner, Larry Cohen, Peter D Donnelly, Robert Muggah, Rachel Davis, Anna Realini, Berit Kieselbach, Lori Snyder MacGregor, et al. 2016. Transforming our world: implementing the 2030 agenda through sustainable development goal indicators. *Journal of public health policy*, 37(1):13–31.

Gianna Marcato and Eva-Maria Thüne. 2002. Gender and female visibility in Italian. *Gender across languages: The linguistic representation of women and men*, 2:187–217.

Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. English Gigaword fifth edition, 2011. *Linguistic Data Consortium, Philadelphia, PA, USA*.

Alma Sabatini. 1985. Occupational titles in Italian: Changing the sexist usage. In *Sprachwandel und feministische Sprachpolitik: Internationale Perspektiven*, pages 64–75. Springer.

Milan Straka, Jan Hajic, and Jana Straková. 2016. UDPipe: Trainable Pipeline for Processing CoNLL-U Files Performing Tokenization, Morphological Analysis, POS Tagging and Parsing. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016*, Portorož,Slovenia, 5. European Language Resources Association (ELRA).

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85. Publisher: ACM New York, NY, USA.

Deep Learning Representations in Automatic Misogyny Identification: What Do We Gain and What Do We Miss?

Elisabetta Fersini, Luca Rosato, Antonio Candelieri, Francesco Archetti, Enza Messina
University of Milano-Bicocca, Milan, Italy
{elisabetta.fersini, antonio.candelieri}@unimib.it
{francesco.archetti, enza.messina}@unimib.it,
l.rosato1@campus.unimib.it

Abstract

In this paper, we address the problem of automatic misogyny identification focusing on understanding the representation capabilities of widely adopted embeddings and addressing the problem of unintended bias. The proposed framework, grounded on Sentence Embeddings and Multi-Objective Bayesian Optimization, has been validated on an Italian dataset. We highlight capabilities and weaknesses related to the use of pre-trained language, as well as the contribution of Bayesian Optimization for mitigating the problem of biased predictions.

1 Introduction

Nowadays, although women, girls and teenagers have a strong presence in online social environments, they are strongly exposed to hateful comments. In 2021, a survey provided by the Pew Research Center has shown that females are targeted for severe types of online gender-based attacks[1]: women are more likely than men to report having been sexually harassed online (16% vs. 5%) or stalked (13% vs. 9%). These phenomena can be found under the umbrella of online misogyny, which can be generally defined as hate, violence or prejudice against women (Ging and Siapera, 2018).

2 State of the Art

In order to counter online misogyny, several computational approaches have been presented in the literature ranging from natural language processing models to machine learning classifiers, denoting quite promising recognition performance. The earliest investigation about computational models for automatic misogyny identification has been presented in Anzovino et al. (2018), where the authors proposed the adoption of several linguistic cues and baseline classifiers for addressing three main problems, i.e., misogyny identification, misogynistic behaviour recognition and target classification. After this seminal paper, several approaches have been presented in the literature distinguishing them according to the feature representations that have considered for representing the textual contents and the machine learning models adopted as classifiers. Most of the approaches experimented a high-level representation of the word and/or sentence (García-Díaz et al., 2021; Pamungkas et al., 2020; Farrell et al., 2020; Lees et al., 2020), coupled with fine-tuning, while few of them adopted shallow models or trained deep architectures from scratch (Fabrizi, 2020; Ou and Li, 2020; da Silva and Roman, 2020; El Abassi and Nisioi, 2020; Koufakou et al., 2020).

Recently, an increasing interest has been focused on the problem of unintended bias (Dixon et al., 2018). In particular, it is important to focus on a given error induced by the training data, i.e., the bias injected in the model by a set of *identity terms* that are frequently associated to the misogynous class. For example, the term *women*, if frequently used in misogynous messages, would lead most of the supervised classification models to over-generalization and to disproportionately associate this identity term to the misogynous label. To this purpose, only few approaches have been dedicated to the unintended bias problem for misogyny identification (Nozza et al., 2019; Lees et al., 2020; Gencoglu, 2020; Zueva et al., 2020), denoting a research panorama that is in its infancy. Although

[1] https://www.pewresearch.org/internet/2021/01/13/the-state-of-online-harassment/

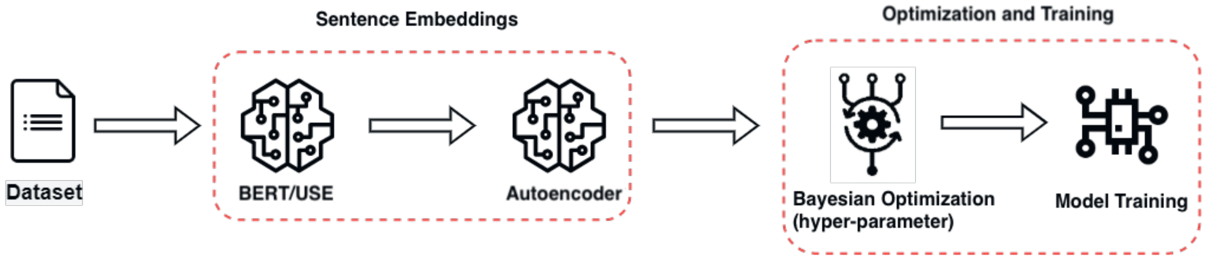


Figure 1: Workflow of the proposed investigation.

the above mentioned approaches represent a fundamental contribution to the problem of automatic misogyny identification in online social environments, they do not focus on two main research questions:
(RQ1) Do embeddings always success when representing different misogyny related problem such as the type of misogyny and the target?
(RQ2) Could classification models be constrained to be less biased by the optimization of their hyper-parameters, therefore having good generalization capabilities also on uncommon expressions?

In this paper, we address the above mentioned open issues by the following main contributions:

- we perform an analysis of capabilities and weaknesses of the widely used state-of-the-art sentence encoders USE and BERT when adopted for misogyny detection;

- we investigate how to reduce the bias of the models by optimizing their hyper-parameters through a multi-objective bayesian optimization strategy.

3 Proposed Framework

In order to address the above mentioned research questions, related to the understanding of weaknesses and capabilities of pre-trained language models for misogyny identification and the reduction of unintended bias, we introduce the framework reported in Figure 1.

3.1 Sentence Embeddings

The proposed approach uses two pre-trained language models to generate a contextual representation of the data. The considered models are based on the *transformer* architecture initially presented in Vaswani et al. (2017). More specifically, the first model is the "small" version of **BERT**, uncased, consisting of 12 stacked *encoders*, 12 parallel *self-attention* and 768 units to represents text. The model is pre-trained on 102 languages, has a dictionary of 110.000 terms and provides a 768-dimensional representation of the text as output. The second model is the multi-language version of **USE** trained on 16 languages, which consists of 6 *stacked encoders*, 8 parallel *self-attention* and 512 units for the text representation. USE provides a 512-dimensional representation of the text, computed as the average over the last encoder's embeddings of each token. The pre-trained BERT and USE models have been *fine-tuned* according to the available misogyny related labels. In order to reduce the dimension of the vector representation given by the fine-tuned pre-trained models and to introduce sparsity to improve the separability of the data, an Autoencoder is used as suggested in (Glorot et al., 2011). The architecture of the Autoencoder is reported in Figura 2.

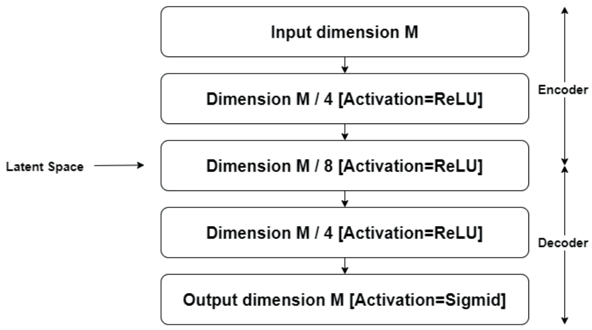


Figure 2: Autoencoder adopted to map the original data in a more compact and sparse representation.

3.2 Training and Optimization

Once the latent representation of a sentence is obtained by means of the Autoencoder, any machine learning model could be adopted to recognize misogynous contents. To this purpose, Support Vector Machines (SVM) have been used, searching for their optimal hyper-parameter settings that are able to ensure the highest recognition performance. However, searching for hyper-parameters that maximize a specific performance metric is a computational expensive black-box optimization process. Due its sample efficiency, Bayesian Optimization (BO), has been adopted. BO works sequentially: each classifier's hyper-parameters to evaluate is chosen by dealing with the exploitation-exploration dilemma. To do this, BO relies on two key components: a *probabilistic surrogate model* approximating the performance metric to optimize - depending on SVM classifiers evaluated so far - and an *acquisition function* (*utility function* suggesting the choice of the next SVM's hyper-parameters to evaluate. The adoption of a probabilistic surrogate model, specifically a Gaussian Process (GP) in this study, allows to estimate the expected value of the performance metric (i.e., GP's predictive mean) and the associated uncertainty (i.e., GP's predictive standard deviation), for any given SVM's hyperparaters configuration. These two estimates are combined into the acquisition function, which implements the exploitation-exploration trade-off mechanism, where exploitation and exploration are associated to the surrogate's predictive mean and standard deviation, respectively. More formally, let $\mathcal{D}_{1:n} = \left\{\left(h^{(i)}, \upsilon^{(i)}\right)\right\}_{i=1,\dots,n}$ be the set of n possible configuration, where $h^{(i)}$ is a d-dimensional vector whose component $h_j^{(i)} \in H_j$ is the value of the j-th hyperparameter of the i-th SVM classifier, and $\upsilon^{(i)}$ is the associated value of the target performance measure. The overall search space H is usually a subspace of the cartesian product of the hyper-parameters's ranges: $H \subseteq H_1 \times \dots \times H_j \times \dots \times H_d$. In this study the search space H is spanned by $d = 2$ hyper-parameters whose values can vary into the following ranges:

- $h_1 \in H_1 := [10^{-1}, 10^5]$, that is the *regularization* hyperparameter C of the SVM classifier (i.e., soft margin SVM)

- $h_2 \in H_2 := [10^{-5}, 10^1]$, that is the hyperparameter γ of the Radial Basis Function kernel of the SVM classifier (i.e., $k(x, x') = e^{-\gamma||x-x'||^2}$)

In this study we consider two different cases (on stratified 10-fold cross-validation):

- tuning the SVM classifier's hyper-parameters to maximize the accuracy, irrespectively to any measure of bias;

- tuning the SVM classifier's hyper-parameters to optimize an objective function aimed at maximizing accuracy and minimizing a *bias-related metric*.

Measuring the Bias In this paper, we measure the model bias by referring to the specific definition of *unintended bias* presented in (Dixon et al., 2018):

> A model contains unintended bias if it performs better for comments containing some particular identity terms than for comments containing others.

In order to measure the level of unintended bias of a given model, *identity terms* (terms related to the *woman* concept) and templates (pre-defined skeleton used to create synthetic samples) are used to generate sentences referred to women, which however can be unreasonably classified as misogynous with high scores. To this purpose, identity terms and templates available for the AMI at Evalita 2020 challenge (Fersini et al., 2020) have been used. Identity terms have been listed using a set of 37 concepts related to "woman", considering both their singular and plural form for the Italian language. Since unintended bias of identity terms cannot be measured on the original dataset set due to class imbalance and highly different identity term contexts, a *synthetic dataset* is generated by means of templates. Following (Nozza et al., 2019), we defined several templates that are filled out with identity terms and with verbs and adjectives that are divided into negative (e.g. hate, inferior) or positive (e.g. love, awesome) forms to convey misogyny or not. Table 1 reports examples of templates. The generated synthetic dataset comprises 3,923 instances, of which 50% misogynous and 50% non-misogynous, where each identity term appears in the same contexts.

Template Examples	Label
`<identity_term>`devono essere protette	Non-Misogynous
`<identity_term>`devono essere torturate	Misogynous
adorare `<identity_term>`	Non-Misogynous
umiliare `<identity_term>`	Misogynous
`<identity_term>`stimabile	Non-Misogynous
`<identity_term>`rivoltante	Misogynous

Table 1: Template examples.

Identity terms, templates and synthetic dataset are available at `https://github.com/MIND-Lab/ItalianBias`.

In order to evaluate the performance of the classification in terms of bias, an AUC-related measure has been used (AUC_{final}). In what follow, the higher is the AUC_{final}, the lower is the bias of the model. In particular, a weighted combination of AUC estimated on the raw dataset AUC_{raw} (original tweets) and three per-term AUC-based scores computed on the synthetic dataset ($AUC_{Subgroup}$, AUC_{BPSN}, AUC_{BNSP}) is adopted (Borkan et al., 2019). Let s be an identity-term (e.g. "donna" and "moglie") and N be the total number of identity-terms, the AUC_{final} is defined as:

$$\begin{aligned} AUC_{final} &= \tfrac{1}{2} AUC_{raw} + \\ &+ \tfrac{1}{2N} \Big[\textstyle\sum_s AUC_{subgroup}(s) \\ &+ \textstyle\sum_s AUC_{BPSN}(s) \\ &+ \textstyle\sum_s AUC_{BNSP}(s) \Big] \end{aligned} \tag{1}$$

where:

- $AUC_{Subgroup}(s)$: computes AUC only on the data within the subgroup containing a given identity term s. This represents model understanding and separability within the subgroup itself. A low value means that the model does not distinguish properly misogynous and non-misogynous comments containing a give identity term s.

- $AUC_{BPSN}(s)$: Background Positive Subgroup Negative (BPSN) estimates AUC on the misogynous examples using the background and the non-misogynous examples belonging the subgroup. A low value means that the model mislead non-misogynous examples that mention the identity-term with misogynous examples that do not, likely meaning that the model predicts higher misogynous scores than it should for non-misogynous examples mentioning the identity-term.

- $AUC_{BNSP}(s)$: Background Negative Subgroup Positive (BNSP) calculates AUC on the non-misogynous examples from the background and the misogynous examples from the subgroup. A low value means that the model confuses misogynous examples that mention the identity with non-misogynous examples that do not, likely meaning that the model predicts lower misogynous scores than it should for misogynous examples mentioning the identity.

4 Experimental Investigation

In this section we report the experimental investigation performed on the Italian version of the Automatic Misogyny Detection (AMI) dataset (Fersini et al., 2020), comparing the results obtained with the proposed framework with the ones obtained by the baseline model (i.e. SVM trained on a TF-IDF representation). The AMI dataset is composed of 5,000 tweets, labelled according to "misogyny" (i.e., indicating if a Tweet is misogynous or not), "misogyny category" (i.e., Stereotype&Objectification, Dominance, Derailing, Sexual Harassment&Threats of Violence, Discredit) and "target" (i.e., individual or generic).

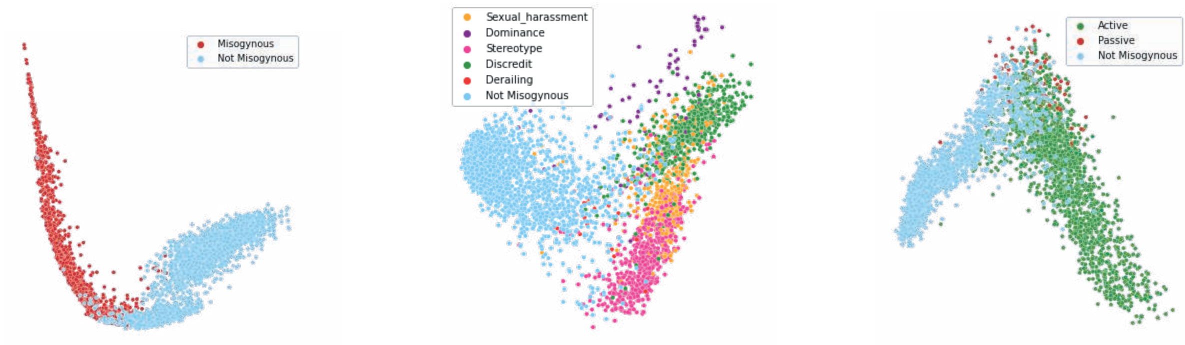
Regarding the first research question **(RQ1)**, we tuned the SVM classifier's hyper-parameters to maximize only the performance measure related to each label (i.e. Accuracy for misogyny labels, F-Measure for category and target labels). First of all, we reported in Table 2 the results comparing different models. It can be easily noted that, although BERT and USE allow SVM to achieve better performance than TFIDF, there is no difference between them achieving similar results. Moreover, while the recognition performance on the misogyny labels are satisfactory, the capabilities on discriminating the misogyny category and the target are still far from being acceptable.

In order to understand if the low performance can be due to the embedding, we investigated the class overlapping originated by USE and BERT. We report in Figure 3, a 2D representation of the embeddings obtained by USE (similar results have been obtained for BERT). We can immediately highlight that while the embeddings tuned for recognizing misogyny are quite distinguishable between misogynous and not misogynous tweets, for the category and target embeddings there is an overlapping among the classes. This makes the learned representations not ready for being used to recognize the specific form of misogyny and the subject of misogynous comments.

Regarding the second research question **(RQ2)**, we determined the SVM optimal hyper-parameters to maximize both $Accuracy$ and AUC_{final} (i.e. the bias-related metric). In order

	Baseline (TFIDF + Opt. SVM)	OUR (BERT + Opt. SVM)	OUR (USE + Opt. SVM)	Absolute Improvement
Misogyny [Accuracy]	0.8390	<u>0.8670</u>	0.8640	+2.8%
Misogyny Category [F-measure]	0.5427	0.5988	<u>0.5991</u>	+5.64%
Target [F-measure]	0.4217	<u>0.4599</u>	0.4537	+3.82%

Table 2: Performance comparison of different approaches. Underlined numbers denote the best result.



(a) Misogyny Embedding.

(b) Category Embedding.

(c) Target Embedding.

Figure 3: 2D embedding representation obtained by USE.

to guarantee the use different set of tokens for the hyper-parameter search and the inference phase, the synthetic samples have been split in training and testing. We compare in Table 3, the AUC_{final} values estimated on biased and unbiased SVM. We can easily note that the Unbiased SVM leads to maintain constant the Accuracy, but improve the generalization capabilities of the embeddings given by the AUC_{final} values, denoting a slightly better results for USE. This means that the SVM hyper-parameter optimization, with respect to both performance measures, leads to promising unbiased models. This ensures ensure good recognition capabilities on both common expressions (typically used on Twitter) and on uncommon comments (synthetic data). The obtained results also suggest that an SVM trained using the USE embedding is more keen to adapt the hyper-parameters to reduce its bias during training and inference.

		Biased SVM	Unbiased SVM
TFIDF	$Accuracy$	0.8390	0.8390
	AUC_{final}	0.6910	0.6950
BERT	$Accuracy$	0.8679	0.8679
	AUC_{final}	0.7197	0.7211
USE	$Accuracy$	0.8640	0.8640
	AUC_{final}	0.7181	**0.7430**

Table 3: Generalizaion capabilites on biased and unbiased models.

5 Conclusions and Future Work

In this paper we have investigated the capabilities and weaknesses of pre-trained language models, as well as the problem of the unintended bias when addressing the automatic misogyny identification for the Italian language. The proposed investigation has highlighted that, while pre-trained embeddings are able to distinguish misogynous and not misogynous comments, they still have poor discrimination capabilities related to the type of misogyny and its target. Regarding the unintended bias problem, it has been shown that an hyper-parameter search guided by Bayesian Optimization can lead to debiased models with good recognition generalization capabilities. As future work, we will investigate explainable AI techniques aimed at generating a feature score that is directly proportional to the feature's effect on inducing bias in the prediction model.

References

Maria Anzovino, Elisabetta Fersini, and Paolo Rosso. 2018. Automatic Identification and Classification of Misogynistic Language on Twitter. In *International Conference on Applications of Natural Language to Information Systems*, pages 57–64. Springer.

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum

Thain, and Lucy Vasserman. 2019. Nuanced Metrics for Measuring Unintended Bias with Real Data for Text Classification. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 491–500.

Adriano dos S. R. da Silva and Norton T. Roman. 2020. No Place For Hate Speech @ AMI: Convolutional Neural Network and Word Embedding for the Identification of Misogyny in Italian. In *Proceedings of Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020)*, Bologna, Italy. CEUR.org.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and Mitigating Unintended Bias in Text Classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73. ACM.

Samer El Abassi and Sergiu Nisioi. 2020. MDD@AMI: Vanilla Classifiers for Misogyny Identification. In *Proceedings of Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020)*, Bologna, Italy. CEUR.org.

Samuel Fabrizi. 2020. fabsam @ AMI: a Convolutional Neural Network approach. In *Proceedings of Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020)*, Bologna, Italy. CEUR.org.

Tracie Farrell, Oscar Araque, Miriam Fernandez, and Harith Alani. 2020. On the use of Jargon and Word Embeddings to Explore Subculture within the Reddit's Manosphere. In *12th ACM Conference on Web Science*, pages 221–230.

Elisabetta Fersini, Debora Nozza, and Paolo Rosso. 2020. AMI@ EVALITA2020: Automatic Misogyny Identification. In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020)*. CEUR.org.

José Antonio García-Díaz, Mar Cánovas-García, Ricardo Colomo-Palacios, and Rafael Valencia-García. 2021. Detecting Misogyny in Spanish Tweets. An Approach based on Linguistics Features and Word embeddings. *Future Generation Computer Systems*, 114:506–518.

Oguzhan Gencoglu. 2020. Cyberbullying detection with fairness constraints. *IEEE Internet Computing*, 25(1):20–29.

Debbie Ging and Eugenia Siapera. 2018. Special issue on online misogyny. *Feminist Media Studies*, 18(4):515–524.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep Sparse Rectifier Neural Networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings.

Anna Koufakou, Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. 2020. HurtBERT: Incorporating Lexical Features with BERT for the Detection of Abusive Language. In *Proceedings of the fourth workshop on online abuse and harms*, pages 34–43.

Alyssa Lees, Jeffrey Sorensen, and Ian Kivlichan. 2020. Jigsaw@ AMI and HaSpeeDe2: Fine-Tuning a Pre-Trained Comment-Domain BERT Model. In *Proceedings of Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020), Bologna, Italy. CEUR. org*.

Debora Nozza, Claudia Volpetti, and Elisabetta Fersini. 2019. Unintended Bias in Misogyny Detection. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 149–155.

Xiaozhi Ou and Hongling Li. 2020. YNU_OXZ @ HaSpeeDe 2 and AMI : XLM-RoBERTa with Ordered Neurons LSTM for Classification Task at EVALITA 2020. In *Proceedings of Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020)*, Bologna, Italy. CEUR.org.

Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. 2020. Misogyny detection in twitter: a multilingual and cross-domain study. *Information Processing & Management*, 57(6):102360.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Nadezhda Zueva, Madina Kabirova, and Pavel Kalaidin. 2020. Reducing Unintended Identity Bias in Russian Hate Speech Detection. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 65–69.

An Obligations Extraction System for Heterogeneous Legal Documents: Building and Evaluating Data and Model

Maria Iacono, Laura Rossi, Paolo Dangelo, Andrea Tesei, Lorenzo De Mattei
Aptus.AI / Pisa, Italy
{maria,laura,paolo,andrea,lorenzo}@aptus.ai

Abstract

A system that extracts obligations automatically from heterogeneous regulations could be of great help for a variety of stakeholders including financial institutions. In order to reach this goal, we propose a methodology to build a training set of regulations written in Italian coming from a set of different legal sources and a system based on a Transformer language model to solve this task. More importantly, we deep dive into the process of human and machine-learned annotations by carrying out both quantitative and manual evaluations of both of them.

1 Introduction

Compliance practitioners in financial intuitions are overburdened by the high volume of upcoming regulations coming from different legal sources, such as the European Union, National legislation, central banks and independent administrative authorities sources, to name a few. Part of the compliance offices work consists of extracting obligations from this vast amount of regulations to trigger compliance processes. It is worth noting that extracting obligations from such a big amount of regulations is tedious and repetitive work. In this scenario having systems to automate this process might be very useful to cut down the costs. Machine Learning (ML) and Natural Language Processing (NLP) may come in help. However, given the variety of legal sources, training this kind of system is a complex activity because it requires a sufficient amount of annotated data, which are expensive especially if the annotations require legal domain experts.

The obligations extraction topic has been already studied with different approaches. Bartolini et al. (2004) used a shallow syntactic parser and hand-crafted rules to automatically classify laws paragraphs according to their regulatory content and extract relevant text fragments corresponding to specific semantic roles. Similarly Sleimi et al. (2018) represent automatically legal texts semantics using an RDF schema with a system based on a dependency parser and hand-crafted rules. Sleimi et al. (2019) used the same representation to build a question-answering system with a focus on obligations. Biagioli et al. (2005) represent law paragraphs using Bag of words either with TF or TF-IDF weighting (Salton and Buckley, 1988) and used Support Vector Machines (SVM) to classify each paragraph as a type of provisioning including obligations. A similar approach is adopted by Francesconi and Passerini (2007): they classify legislative texts paragraphs according to the proposed provision model. They represent them in a similar way as (Biagioli et al., 2005) and use two learning algorithms: Naive Bayes and SVM. Sleimi et al. (2020), propose to address the problem of the complexity of regulatory texts by writing them following a set of standard templates which could be easily parsed.

Contributions In this work we offer four main contributions. (i) We propose a methodology for building training corpora relying on non-expert annotators and we apply this methodology on a set of heterogeneous regulations written in Italian, coming from a set of different legal sources. (ii) We assess the quality of the introduced methodology relying on an inter-annotator agreement score and we carry out an error analysis to highlight if and when expert annotators are required. (iii) We use the dataset produced to train and test an obligations classification system based on neural networks as this approach has been proven to provides state of the art results for several Italian classification tasks (De Mattei et al., 2018; Cimino et al., 2018; Occhipinti et al., 2020). (v) We conduct a manual error analysis to investigate the pros and the limitations of the mentioned system.

2 Task Description

The task we tackle consists of classifying regulations clauses either as obligations or not. By obligation, we mean, from a juridical point of view, a legal constraint imposed by law and addressed to a juridical person.

Being interested in developing a system that supports financial institutions, we distinguish two categories of obligations, classifying them as relevant or irrelevant for financial institutions. Then each clause can be classified in one out of the following three categories: (i) `not obligation`, (ii) `relevant obligation` and (iii) `not relevant obligation`. This classification schema allows practitioners to retrieve in one click all the obligations or the relevant only so that they can decide whether to have a complete overview of the laws they are consulting or to focus only on the obligations that actually affect their institutions.

To distinguish the two categories, we look at the subject to whom the obligation is addressed: if it is a public institution, we classify it as an irrelevant obligation, in all other cases as a relevant obligation. This simplification applied to the classification criterion may seem extreme since it implies that any type of obligation not addressed to a public institution must be considered relevant for a financial institution. However, we believe that applying this distinction is a good strategy because the documents we analyze are already filtered, i.e., they belong to a category of laws that impact financial institutions. Consequently, within them, if an obligation is not directed at a public institution it will almost certainly be directed somehow to financial institutions.

2.1 Special Cases

Legal jargon is not merely a tool used for argumentation or narrative, but a constitutive element of the law. Consequently, the structure of legal texts has particular characteristics that must respond to precise and predictable patterns. Despite this, there are cases in which the language can be ambiguous. Since our goal is to build a dataset in line with compliance practitioners expectations we analyzed some special cases with a group of experts in order to provide clear guidelines to annotators.

One such case is when an obligation is expressed indirectly, for example through the formulation of a right. If an article talks about rights of any kind, it assumes that those rights must be respected. So, for example, the right of a client in terms of obtaining a loan (client's point of view) corresponds to a duty of the bank, which is obliged to grant it if the client has what it takes (bank's point of view). Similarly, an employee's right to go on vacation means that the employer must guarantee vacation days. For this reason, in deciding how to classify a part of a law, in addition to the interpretation by the annotator, the concept of "priority" comes into play. Since our application is designed to support financial institutions, our priority is to highlight the obligations that they must take into account in order not to risk penalties. Consequently, if a sentence represents both a right for one subject and duty for another, we prioritize the obligation in classifying it.

Another case where the priority factor comes into play is that of clauses that contain both relevant and irrelevant obligations. In these cases, since we cannot break the clause down into several parts, we give priority to the relevant obligation. In terms of risk, it is better to classify an irrelevant obligation as relevant, rather than the other way around.

In addition, we have to consider that obligations may be reported implicitly. For example, if a person can perform an action only under certain conditions, it is implied that those conditions can be interpreted as obligations. According to this principle, we do not classify a sentence such as "Spectators may enter the theatre" as an obligation. On the contrary, we do so when a condition is added, as in the case of the sentence "Spectators may enter the theatre only if they have the ticket."

Even if we, as readers, do not pay attention to it, normative texts often contain implicit information that readers are naturally able to trace through reading, such as an implied subject, or a reference to another part of the document or to an external document. Unlike a reader, an automatic classifier, not having provided with enough context, may en-counter difficulties in handling this kind of case.

3 Data Annotation

We extracted the dataset from Daitomic[1], a product that automatically collects legal documents from a wide variety of legal sources, represents automatically them accordingly to the Akoma Ntoso standard (Palmirani and Vitali, 2011) and makes them available through a dedicated User Interface. The adoption of Akoma Ntoso lets us represent the structure of heterogeneous legal texts in a unified format that makes us able to apply the same operations on very different kind of poorly encoded documents such as PDF, HTML and DOCX files.

The corpus has been manually labelled by three trained annotators with no previous background in legal domain and contains 71 regulations for a total of 10.628 clauses. We selected regulations that touch heterogeneous topics such as data privacy, financial risk, tax compliance and many more but all of them are known to be relevant for financial institutions. In order to deal with the problem of heterogeneity of normative sources, we found it appropriate to take texts from different sources, so that we could train the model in a balanced way. In particular, we extracted the texts from thirty of the most important regulatory sources for Italian financial institutions, including Gazzetta Ufficiale Italiana, EUR-Lex, Consob, Banca d'Italia and many more. From these sources, we selected texts of different types: acts, regulations, decisions, directives, communications, statutes, and more. In this way, we created a very heterogeneous dataset that can be considered representative of the wide variety of existing regulations.

The annotations were carried out directly from the graphical user interface of the Daitomic application, which allows, within the consultation section, to mark the requirements present in the law and to classify them as relevant or not relevant. The application texts are already structured, so they present a tree structure divided into chapters, articles, paragraphs, clauses, etc, where we annotated the smallest parts, i.e. clauses. Each clause is flanked by a sidebar, clicking on which automatically opens the pop-up shown in Figure 1, which allows the annotators to choose the label that they consider most appropriate. As a result of this choice, the sidebar will turn light blue if the obligation is classified as relevant to financial institutions, and dark blue if it is not relevant.

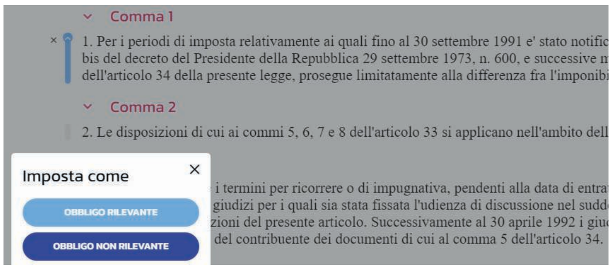


Figure 1: Pop-up for setting the label of the obligation.

We picked four of the annotated laws containing as many as 2189 clauses to be annotated by all three annotators.

4 Annotations Evaluation

We used the part of the dataset annotated by all three annotators in order to calculate the inter-annotator agreement (IAA). Using Krippendorff's Alpha reliability, we computed IAA in two different ways, at first checking only whether they had classified the sentences as obligations or non-obligations, then taking into account their choices in distinguishing obligations between relevant and non-relevant. The resulting IAA is 0.58 considering the distinction between relevant and not relevant but increases to 0.70 if no such distinction is applied.

In order to better understand these results we carried out a manual analysis from which turned out that most cases of disagreement are of two kinds (two examples are reported in Table 1). The lack of agreement between annotators can be primarily attributed to the fact that there is often no explicitly expressed subject in a clause, either because it is expressed in the preceding clauses or because it is intuitable from the context, as we can see in the first example. Another frequent reason for disagreement is surely the fact that our annotators, not being experts in the legal field, not always are able to understand the kind of subject to which the obligation is referred, as in the second example. In such cases, expert annotators might be more reliable.

5 Automatic Classifier

We also used the dataset we built to train an automatic classifier. We split the dataset into training (90%) and test (10%) sets. As a learning

[1] https://www.daitomic.com/

Annotator 1	Annotator 2	Annotator 3	text
not relevant	relevant	relevant	I contratti di assicurazione di cui al comma 1, lettera b), sono corredati da un regolamento, redatto in base alle direttive impartite dalla COVIP [...] *en:[The insurance contracts referred to in paragraph 1, letter b), are accompanied by a regulation, drawn up on the basis of the directives issued by COVIP [...]]*
relevant	relevant	not relevant	Il soggetto incaricato del collocamento nel territorio dello Stato provvede altresi' agli adempimenti stabiliti [...] *en:[The person in charge of placement in the territory of the The State also provides for the established obligations [...]]*

Table 1: Example of disagreement among annotators. Correct classifications are shown in blue while incorrect classifications are shown in red.

	Precision	Recall	F-Score
Not Obligations	0.96	0.98	0.97
Relevant Obligations	0.67	0.63	0.65
Not Relevant Obligations	0.84	0.76	0.80

Table 2: System performances evaluation on the test set

	Precision	Recall	F-Score
Not Obligations	0.96	0.98	0.97
Obligations	0.95	0.87	0.91

Table 3: System performances evaluation on the test set with no distinguish between relevant and not relevant obligations

model, we used UmBERTo[2], an Italian pretrained Language Model trained by Musixmatch based on Roberta architecture (Liu et al., 2019), which has been recently proved to provide state of the art performances for other Italian tasks (Occhipinti et al., 2020; Sarti, 2020; Giorgioni et al., 2020). This language model has 12-layer, 768-hidden, 12-heads, 110M parameters. On top of the language model, we added a ReLU classifier (Nair and Hinton, 2010). All the model's weights has been updated during fine-tuning. We applied dropout (Srivastava et al., 2014) with probability 0.1 to both the attention and the hidden layers. We used Cross-Entropy as a loss function and we trained the system until early-stop at epoch 6. The performances obtained on the test set are reported in Table 2. The system performances are fairly good if compared to IAA but not enough reliable to be used in real-world scenarios. However if we evaluate the system without considering the difference between not relevant and relevant obligations (Table 3) we observe much more accurate results suggesting that the systems, similarly to the annotators, performs well in identifying obligations, but struggles in distinguishing between relevant and not relevant obligations.

6 Human vs Automatic Classification

In order to better understand the model capabilities, we ran a manual error analysis, comparing human annotations against automatic classifications on the test set. We identified some categories of typical errors and reported some examples in Table 4. In some cases, the errors of the model are attributable to the non-explicit subject, which the human annotator can derive from the context, as can be seen in the first example, where it is not explicitly specified who should enter the data in the communication. Looking at the second example, we can see a sentence whose main message is the expression of a right, in this case, the right to access a certain file. However, access to the file is allowed only under certain temporal conditions (*at the conclusion of the appeal procedure*), so behind that right is hidden a relevant obligation. Unfortu-

[2]`https://github.com/musixmatchresearch/umberto`

Human	Machine	text
not relevant	relevant	Nella comunicazione di avvio di cui al comma 2 sono indicati l'oggetto del procedimento, gli elementi acquisiti d'ufficio [...] *en:[In the communication of initiation referred to in paragraph 2 are indicated the subject of the procedure, the elements acquired ex officio [...]]*
relevant	none	L'accesso al fascicolo è consentito a conclusione della procedura di interpello ai fini della tutela in sede giurisdizionale. *en:[Access to the file is granted at the conclusion of the appeal procedure for judicial protection purposes.]*
relevant	none	E' considerata ingannevole la pubblicità', che, in quanto suscettibile di raggiungere bambini ed adolescenti, può', anche indirettamente, minacciare la loro sicurezza. *en:[Advertising that is likely to reach children and adolescents and that may even indirectly threaten their safety is considered misleading.]*
relevant	not relevant	Le amministrazioni interessate provvedono agli adempimenti previsti dal presente decreto con le risorse umane, finanziarie e strumentali disponibili [...]. *en:[The administrations involved shall carry out the obligations provided for in this decree with the human, financial and instrumental resources available.[...]]*
relevant	none	Il presente decreto reca le disposizioni di attuazione dell'articolo 1 del decreto legge 6 dicembre 2011, n. 201, convertito, con modificazioni, dalla legge 22 dicembre 2011, n. 214 [...]. *en:[This decree contains the provisions for the implementation of article 1 of Law Decree no. 201 of December 6, 2011, converted, with amendments, by Law no. 214 of December 22, 2011 [...]]*

Table 4: Example of disagreement between manual (*Human*) and automatic (*Machine*) annotations. Correct classifications are shown in blue while incorrect classifications are shown in red.

nately in these cases, the model is often wrong. Another difficult case to handle is the one shown in the third example in Table 4. This is a sentence that apparently contains simple information: advertising is considered deceptive if it can threaten the safety of children. But behind this message lies an obligation on advertisers to avoid such a situation. Again, the obligation is not explicit, so it is quite understandable that the model could be wrong. Finally, the last two examples show human errors, and it was noted with some interest that where annotators make errors due to distraction or misunderstanding, the model often classifies correctly.

7 Conclusions

In this work we propose a methodology for building training corpora for obligations classification, based on annotations performed by non-experts. We apply this methodology to a set of heterogeneous regulations from a collection of different legal sources. IAA and a manual error analysis highlight that human annotation is in general prone to errors and that non-expert annotators struggle to distinguish between relevant and not relevant obligations. The dataset produced has been used to train and test an obligations classification system based on state-of-the-art pretrained language models. We conduct both an automatic evaluation and a manual error analysis from which turned out that the system, similarly to human annotators, has good performances in recognizing obligations but struggles in distinguish between relevant and not. As future works, we plan to involve domain-expert annotators to evaluate if their contribution can improve the quality of the data and of the model. Also, we will explore techniques to provide more context to the classifier in order to improve the performances on clauses in which the subject is implied.

References

Roberto Bartolini, Alessandro Lenci, Simonetta Montemagni, Vito Pirrelli, and Claudia Soria. 2004. Automatic classification and analysis of provisions in italian legal texts: a case study. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 593–604. Springer.

Carlo Biagioli, Enrico Francesconi, Andrea Passerini, Simonetta Montemagni, and Claudia Soria. 2005. Automatic semantics extraction in law documents. In *Proceedings of the 10th international conference on Artificial intelligence and law*, pages 133–140.

Andrea Cimino, Lorenzo De Mattei, and Felice Dell'Orletta. 2018. Multi-task learning in deep neural networks at evalita 2018. *Proceedings of the Wvaluation Campaign of Natural Language Processing and Speech tools for Italian*, pages 86–95.

Lorenzo De Mattei, Andrea Cimino, and Felice Dell'Orletta. 2018. Multi-task learning in deep neural network for sentiment polarity and irony classification. In *NL4AI@ AI* IA*, pages 76–82.

Enrico Francesconi and Andrea Passerini. 2007. Automatic classification of provisions in legislative texts. *Artificial Intelligence and Law*, 15(1):1–17.

Simone Giorgioni, Marcello Politi, Samir Salman, Roberto Basili, and Danilo Croce. 2020. Unitor@ sardistance2020: Combining transformer-based architectures and transfer learning for robust stance detection. In *EVALITA*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *ICML*.

Daniela Occhipinti, Andrea Tesei, Maria Iacono, Carlo Aliprandi, Lorenzo De Mattei, and Aptus AI. 2020. Italianlp@ tag-it: Umberto for author profiling at tag-it 2020. In *Proceedings of Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020), Online. CEUR. org*.

Monica Palmirani and Fabio Vitali, 2011. *Akoma-Ntoso for Legal Documents*, pages 75–100. Springer Netherlands, Dordrecht.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523.

Gabriele Sarti. 2020. Umberto-mtsa@ accompl-it: Improving complexity and acceptability prediction with multi-task learning on self-supervised annotations. *arXiv preprint arXiv:2011.05197*.

Amin Sleimi, Nicolas Sannier, Mehrdad Sabetzadeh, Lionel Briand, and John Dann. 2018. Automated extraction of semantic legal metadata using natural language processing. In *2018 IEEE 26th International Requirements Engineering Conference (RE)*, pages 124–135. IEEE.

Amin Sleimi, Marcello Ceci, Nicolas Sannier, Mehrdad Sabetzadeh, Lionel Briand, and John Dann. 2019. A query system for extracting requirements-related information from legal texts. In *2019 IEEE 27th International Requirements Engineering Conference (RE)*, pages 319–329. IEEE.

Amin Sleimi, Marcello Ceci, Mehrdad Sabetzadeh, Lionel C Briand, and John Dann. 2020. Automated recommendation of templates for legal requirements. In *2020 IEEE 28th International Requirements Engineering Conference (RE)*, pages 158–168. IEEE.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

FONTI 4.0: Evaluating Speech-to-Text Automatic Transcription of Digitized Historical Oral Sources

Roberta Bianca Luzietti[1], **Niccolò Pretto**[1], **Frédéric Kaplan**[2], **Alain Dufaux**[2] **and Sergio Canazza**[1]
1. University of Padua, Italy
2. École Polytechnique Fédérale de Lausanne
robertabianca.luzietti@unipd.it,
{niccolo.pretto,canazza}@dei.unipd.it,
{frederic.kaplan,alain.dufaux}@epfl.ch

Abstract

Conducting "manual" transcriptions and analyses is unsustainable for most historical oral archives because they require a remarkable amount of funds and time. The FONTI 4.0 project aims at exploring the suitability of automatic transcription and information extraction technologies for making historical oral sources available. In this work, we conducted an experiment to test the performance of two commercial speech-to-text services (Google Cloud Speech-to-text and Amazon Transcribe) on digitized oral sources. We created an eight-hour corpus made of manually transcribed and annotated historical speech recordings in TEI format. The results clearly show how audio quality and disturbing elements (e.g., overlaps, foreign words, etc.) impact on the automatic transcription, showing what needs to be improved for implementing an unsupervised transcription chain.

1 Introduction

FONTI 4.0[1] is a project aiming at exploring the suitability of automatic transcription and analysis tools for the preservation of historical oral sources recorded on analog carriers, in particular magnetic tapes. The digitization of an audio archive is a long and expensive task that can require several years. Furthermore, the content of audio recordings needs to be listened and cataloged for making audio recordings retrievable. Archives composed by hundreds or thousands of hours of audio require a huge amount of time, people and funds for making the content accessible and preventing their exploitation. Therefore, automatizing the transcription and the analysis task could drastically reduce the time for making digitized audio recordings accessible.

The project consists in a transcription-chain (T-chain), firstly defined in (van Hessen et al., 2020), that differs in two main aspects: (a) in FONTI 4.0, the transcription obtained with speech-to-text (STT) algorithms should not be corrected by human; (b) an additional restoration step could be required for digitized audio recordings. Furthermore, differently from STT evaluation experiments conducted by (Moore et al., 2019; Kostuchenko et al., 2019; Filippidou and Moussiades, 2020), we decided to employ two commercial software, namely Google Cloud Platform and Amazon Web Services, to test their ability to transcribe historical analog recordings, and to eventually include in our pipeline.

During the digitization process, speed and equalization errors can occur, especially when different speed and equalization configurations are used in different part of the same tape (Pretto et al., 2020). This leads to distortions of the recorded signal that becomes unlistenable. By using the correction workflow and digital filters described in (Pretto et al., 2021a; Pretto et al., 2021b) these errors can be corrected and at least parts of the signal can be saved. This task is essential for making the speech signal suitable for STT algorithms. This paper aims at evaluating the transcription performance of two commercial software on a real use case and identifying potential problems or limitations concerning peculiarities of analog audio recordings. Section 2 describes the corpus, used

[1] csc.dei.unipd.it/fonti40en/ (last accessed September 2nd, 2021)

as ground-truth for the experiment. Section 3 outlines the methodology adopted for this experiment, whereas results are reported in Section 4. Finally, Section 5 presents the authors' conclusions.

2 Corpus

The *Cinema & Civiltà* (C&C) corpus was conceived within the FONTI 4.0 as ground-truth for evaluating the performance of STT services on a real case study. To build the corpus, we transcribed speech recorded on four magnetic tapes made available by the Giorgio Cini Foundation in Venice and digitized at the Centro di Sonologia Computazionale - CSC (Canazza and De Poli, 2020). The recordings are parts of the *Cinema & Civiltà* conference for the awarding of the San Giorgio prize, part of the Venice Film Festival, that took place between the 7^{th} and 9^{th} of September 1959, attended by important figures of the history of cinema such as Roberto Rossellini and representatives of the Italian literary critics such as Vittore Branca. Each reel of magnetic tape is composed of two sides: each side counting 60 minutes of recorded speech for a total of eight hours of recording. The C&C corpus is also a multilingual corpus of 64,930 tokens and three subcorpora: Italian 49,772 tokens, French 9,555 tokens (L1 and L2), and Spanish 5,603 tokens. This corpus was manually transcribed and annotated as described in the following subsections and is available at this link[2].

2.1 Transcription

Defining the methodology for the transcription is an important step for the preservation, analysis and access of oral sources. The main difficulty consists in making decisions on how to represent and convey both verbal and non-verbal elements in written form. Because of the absence of a universal standard of transcription (Schorrsidt, 2011), the methodology usually depends on the research aim.

In this research, we decided to complete a verbatim transcription, by reporting every word spoken in the recording including errors, false starts, truncations, and overlaps in Italian, French and Spanish. Using the software ELAN (Lausberg and Sloetjes, 2016), we first segmented audio files extracted from the digitized tapes, making the start and end of each segment coincide with the speaker's turn of talk. Then, we transcribed each segment while listening to the corresponding part of audio in slow motion. Eventually, we opted for employing automatic transcriptions from Google Cloud Speech-to-text (GCS) and Amazon Transcribe (AT)[3], later used in the STT experiment, and correcting the text playing the audio at normal speed. This allowed us to save half the time for each transcription, which previously required a full day of work. Moreover, we were able to retrace and match the identity of the speakers to the voices in the recordings, through the consultation of historical documentation on the conference, and also by comparing voices across the recordings.

2.2 Annotation

The annotation was employed for the addition of important metadata to the C&C corpus regarding different levels of audio quality and the presence of disturbing elements in the recordings. Our methodology is in compliance with the Text Encoding Initiative (TEI) standard guidelines[4] for transcribed spoken material (Burnard and Bauman, 2007). To proceed with the annotation, we first converted the transcription files from the ELAN .eaf into the XML TEI standard using the EXMARaLDA (Schmidt and Wörner, 2014) tool TEI Drop (Schorrsidt, 2011). Subsequently, we used Oxygen[5] to assign TEI tags to the relevant tokens. The list of tags together with a brief description and examples is given below:

<**pause**> marks a pause either between or within utterances in the same segment, **e.g.**: *unica fisionomia. <pause/> Parte dell'architettura*;

<**unclear**> contains a word, phrase, or passage that could not be transcribed with certainty because it is illegible or inaudible in the source, **e.g.**: *gli stessi <unclear reason="inaudible"> strumenti </unclear>, volti agli stessi fini*;

<**gap**> indicates a point where material has been omitted in the transcription because it is inaudible, **e.g.**: *erba che sorgerà <gap reason="inaudible"/> quell'asfalto.*;

[2] DOI: 10.5281/zenodo.5645827

[3] Automatic transcriptions were obtained on the 16th, 17th, 19th and 24th of March 2021.

[4] `tei-c.org/release/doc/tei-p5-doc/en/html/TS.html` (last accessed September 3rd, 2021

[5] `oxygenxml.com` (last accessed September 3rd, 2021)

<**foreign**> identifies a word or phrase as belonging to some language other than that of the surrounding text, **e.g.**: *<foreign xml:lang="fr-FR"> Mesdames, messieurs </foreign>*;

<**shift**> marks the point at which some paralinguistic feature of a series of utterances by any one speaker changes, **e.g.**: *Io credo che questo argomento sia <shift feature="tempo" new="a"/> particolarmente importante <shift feature="tempo" new="normal"/> per vedere*;

<**del**> contains a letter, word, or passage indicated as superfluous by the annotator, in this case it was used for false starts, repetitions and truncations, **e.g.**: *in questo <del type="falseStart"> moden </del> momento* (false start) *momento di <del type="repetition"> di </del> crisi* (repetition) *suggestione di <del type="truncation"> spettaco </del> di spettacolo (truncation)*;

<**anchor**> was used to mark overlaps by attaching an identifier to a point within a text, **e.g.**: *a contatto di un <anchor synch="ovrl6" xml:id="S06"/> pensiero <anchor synch="ovrl6e" xml:id="S06e"/> lo inducono a (interrupted speaker) <anchor xml:id="ovrl6"/> Io non lo vedo. Chi è questo? Chi è questo? <anchor xml:id="ovrl6e"/> (interrupting speaker)*;

<**distinct**> identifies any word or phrase which is regarded as linguistically distinct, as in the case of prosodically unified units, **e.g.**: *staccarsi da <distinct type="pcu"> questa estetica </distinct> e dai pregiudizi*;

<**vocal**> marks any vocalized but not necessarily lexical phenomenon, **e.g.**: *del nostro mondo <vocal> <desc>cough</desc> </vocal> che direi postmoderno.*;

<**incident**> marks any phenomenon or occurrence, not necessarily communicative, for example incidental noises or other events affecting communication, **e.g.**: *è attività creatrice, <incident><desc>noise</desc></incident> ma non propriamente l'artista*;

<**note**> contains notes or citations, and, for the purpose of this research, it was used to annotate the audio quality at the beginning of each segment, **e.g.**: *<note>good </note>*;

Audio quality annotations (<note>) were assigned to each segment using the the following scale (Samar and Metz, 1988):

excellent: speech is completely intelligible;

good: speech is intelligible with the exception of a few words or phrases;

fair: with difficulty, the listener can understand about half the content of the message;

poor: speech is very difficult to understand, only isolated words or phrases are intelligible;

bad: speech is completely unintelligible.

The distribution of words (without punctuation and events) for each audio quality annotation is reported in Table 1.

Scale	it-IT	fr-FR	es-ES	TOT
Excel.	9,075	5,930	4,097	19.102
Good	30,571	2,514	800	33.885
Fair	2,919	83	0	3,002
Poor	1,417	23	0	1,440
TOT	43,984	8,550	4,897	0

Table 1: Number of words (no punctuation nor events) annotated with different audio quality tags.

3 Experiments

The STT experiment consisted in testing the ability of GCS and AT to correctly transcribe historical recordings. Furthermore, we decided to investigate the performance of STT transcriptions obtained from GCS and AT at different levels of audio quality and in presence of disturbing elements in the recordings such as background noise, overlaps, code switching etc. (see Section 2.2).

To analyze the performance of the two STT systems, we developed a Jupyter notebook able to filter the text by language, audio quality, disturbing elements, etc., and select several options, such as tokenization rules. In this experiment, we decided to use only lower case characters, split apostrophes and remove punctuation from both manual and automatic transcriptions. The groundtruth and the resulting transcription of the STT services were canonicalized. The alignment algorithm works on single utterances and minimizes the Levenshtein distance (Jurafsky and Martin, 2008). The obtained metrics were: the number of correct matches (COR) and mismatches, i.e.: deletions (DEL), substitutions (SUB) and insertions (INS), and the word error rate (WER), which is the ratio between the number of mismatches and words in the reference text (Morris et al., 2004). It is important to note that we did not employ this metric to tell how good a system is, but only that one is better than the other (Errattahi et al., 2018).

In order to avoid the introduction of errors not due to the transcription task, we decided not to use the automatic language recognition feature because it could drastically impact on the performance. Therefore, we cut and divided the audio files in different languages and automatically transcribed them separately.

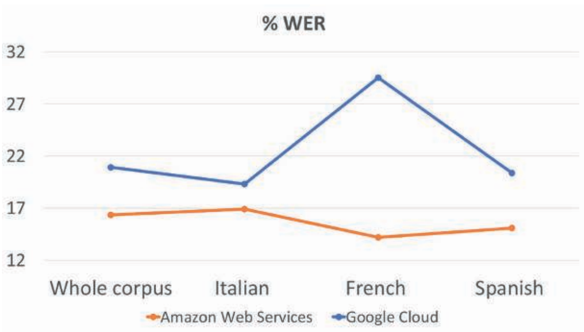


Figure 1: WER of GCS and AT transcriptions on the whole corpus and sub-corpora.

STT	WER	COR	DEL	SUB	INS
AT	16.35%	49,480	2,639	**5,312**	**1,440**
GCS	20.92%	46,510	**5,837**	5,084	1,094

Table 2: Word error rate (WER), Correct matches (COR), deletions (DEL), substitutions (SUB) and insertions (INS) of the Amazon Transcribe (AT) and Google Cloud Speech-to-text (GCS) transcriptions of the overall C&C corpus.

4 Results

In this preliminary work we illustrate and compare mainly WER trends between the two STT systems, calculated on the entire corpus as well as each sub-corpora in relation to audio quality levels and the presence of disturbing elements.

Figure 1 illustrates that the performance of AT are better than GCS in all corpora. The difference between the two systems is small in the Italian sub-corpus, but much wider in the French. A possible explanation could be the presence of L2 speakers of French whose pronunciation could have negatively affected the recognition performance. Nevertheless, it should be also considered that the Italian sub-corpus is more than five times bigger than the French and the Spanish.

STT software performance can be further observed in Table 2: for the transcription of the whole corpus, AT scores a lower WER and finds more correct matches than GCS. On the other hand, deletions in GCS are more than double than in AT, whereas substitutions and insertions are higher in AT than in GCS. In any case, the number of deletions and insertions between AT and GCS are different probably because the two services make use of different language model weights.

Figure 2 shows that transcription performance are very similar in Italian and Spanish with "Excellent" quality, but not in French. For this reason, we cannot impute the bad GCS performance to audio quality. In the Italian sub-corpus, performance are also similar with "Good" quality, but not in the Spanish, where both services performed badly. The negative impact of audio quality is also evident in the French sub-corpus, despite WER values are much higher than Italian.

Results in Figure 3 display the annotated disturbing events found in the C&C corpus that were assumed to negatively affect the performance of STT software in terms of WER. The element that provides the minor disturbance is shift, although the scored WER value for this tag is higher than the one calculated on the overall evaluation. About the other disturbing elements, they show a major impact on the transcription of both STT services. Overall, AT performance is better with most disturbing elements. The only exception is represented by code-switching events in foreign languages for which GCS had a better performance.

5 Conclusion

In this article we conducted a preliminary research experiment testing the ability of STT software to correctly transcribe digitized historical oral sources on magnetic tape. It should be noted, that since this preliminary work has been conducted on a small sample of data, our results are

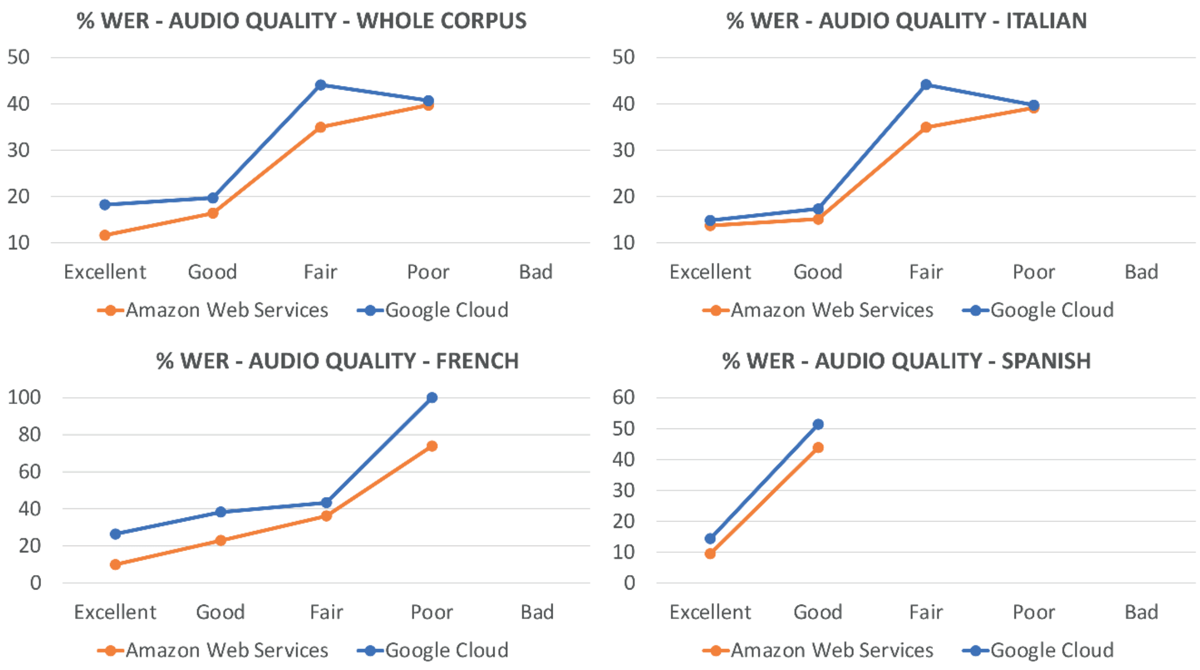


Figure 2: WER of GCS and AT with different audio quality - whole corpus and sub-corpora.

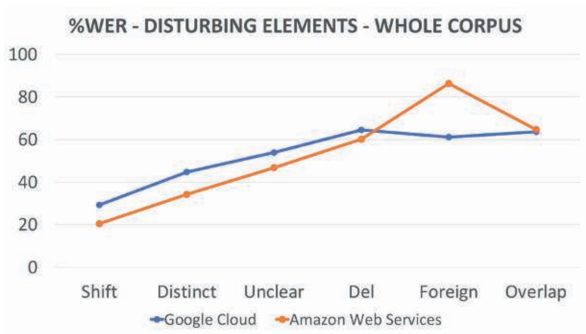


Figure 3: WER of GCS and AT with elements on the whole corpus and sub-corpora.

only indicative of which elements represent the biggest obstacle for STT software performance.

In spite of disturbing elements and the variation of audio quality in the recordings, we demonstrate that with our dataset and in terms of WER, AT performed more accurate transcriptions compared to GCS . On the other hand, GCS was better at recognizing foreign words. Table 2 shows that AT introduces less incorrect words but more insertions and substitutions. This should be taken into consideration when working with automatic information extraction tools (e.g., Named Entity Recognition algorithms) applied to automatic transcriptions. Further analysis should investigate the cause of this trend, to verify if this behavior is also due to alignment or tokenization errors.

With respect to software performance evaluations in relation to variables characterizing analog recordings of speech, we found evidence that audio quality drastically impacts on the number of mismatches. Observations about the incidence of disturbing elements, on the other hand, cannot be generalized since sub-corpora are in three different languages and have three different sizes. Throughout the analyses we noted that the most negative impact on transcription, in terms of the increase of WER, is caused by the presence of some specific recurring elements, i.e.: code-switching (foreign), overlaps and probably even the production of L2 speakers (Figure 3). Nonetheless, given the necessity of preserving historical documents in a more time and cost effective way, we came to the conclusion that researchers working on the preservation of historical recordings will benefit from the use of the T-chain. This is because the reduction by half of the time required for manual transcriptions in slow motion does compensate the lack of accuracy. This means that researchers working on the collection and preservation of oral archives will be able to focus on filling the gap between human and machine output.

Further contributions will be necessary for conducting experiments on L1 and L2 data separately, cross-language testings reducing the Italian subset to the size of the French and Spanish sub-corpora and evaluating the impact of incorrect transcriptions on WER. Language identification through code-switching is another important problem for automatic transcription. Both services recently provided this functionality, but while we are writing this paper, the Google Cloud is still a preview version. As soon as the feature will be available the performance of automatic language recognition algorithms should also be investigated, especially because this feature is essential for automatizing the transcription of entire archives.

Acknowledgments

This paper is produced under the FONTI 4.0 project, financed by resources from the Regional Operational Program co-financed with the European Social Fund 2014-2020 of the Veneto Region. An important acknowledge is due to Fondazione Giorgio Cini, Venice for making available its precious audio material as well as for its help in the recording analysis, and Matteo Pettenò for his contribution in the Jupyter notebook development.

References

Lou Burnard and Syd Bauman, editors, 2007. *TEI P5: Guidelines for Electronic Text Encoding and Interchange*, chapter A Gentle Introduction to XML. Text Encoding Initiative Consortium.

Sergio Canazza and Giovanni De Poli. 2020. Four Decades of Music Research, Creation, and Education at Padua's Centro di Sonologia Computazionale. *Computer Music Journal*, 43(4):58–80, 10.

Rahhal Errattahi, Asmaa El Hannani, and Hassan Ouahmane. 2018. Automatic speech recognition errors detection and correction: A review. *Procedia Computer Science*, 128:32–37.

Foteini Filippidou and Lefteris Moussiades. 2020. A benchmarking of ibm, google and wit automatic speech recognition systems. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 73–82. Springer.

Daniel Jurafsky and James Martin. 2008. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, volume 2. Pearson, Prentice Hall, 02.

Evgeny Kostuchenko, Dariya Novokhrestova, Marina Tirskaya, Alexander Shelupanov, Mikhail Nemirovich-Danchenko, Evgeny Choynzonov, and Lidiya Balatskaya. 2019. The evaluation process automation of phrase and word intelligibility using speech recognition systems. In *International Conference on Speech and Computer*, pages 237–246. Springer.

Hedda Lausberg and Han Sloetjes. 2016. The revised neuroges–elan system: An objective and reliable interdisciplinary analysis tool for nonverbal behavior and gesture. *Behavior research methods*, 48(3):973–993.

Meredith Moore, Michael Saxon, Hemanth Venkateswara, Visar Berisha, and Sethuraman Panchanathan. 2019. Say what? A dataset for exploring the error patterns that two ASR engines make. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2019-Septe:2528–2532.

Andrew Cameron Morris, Viktoria Maier, and Phil Green. 2004. From wer and ril to mer and wil: improved evaluation measures for connected speech recognition. In *Eighth International Conference on Spoken Language Processing*.

Niccolò Pretto, Alessandro Russo, Federica Bressan, Valentina Burini, Antonio Rodà, and Sergio Canazza. 2020. Active preservation of analogue audio documents: A summary of the last seven years of digitization at csc. In *Proceedings of the 17th Sound and Music Computing Conference, SMC20*, pages 394–398, Torino, Italy.

Niccolò Pretto, Nadir Dalla Pozza, Alberto Padoan, Anthony Chmiel, Kurt James Werner, Alessandra Micalizzi, Emery Schubert, Antonio Rodà, Simone Milani, and Sergio Canazza. 2021a. A workflow and novel digital filters for compensating speed and equalization errors on digitized audio open-reel tapes. In *Proceedings of Audio Mostly 2021, AM21*, Trento, Italy.

Niccolò Pretto, Edoardo Micheloni, Anthony Chmiel, Nadir Dalla Pozza, Dario Marinello, Emery Schubert, and Sergio Canazza. 2021b. Multimedia Archives: New Digital Filters to Correct Equalization Errors on Digitized Audio Tapes. *Advances in Multimedia*, 2021:5410218.

Vincent J Samar and Dale Evan Metz. 1988. Criterion validity of speech intelligibility rating-scale procedures for the hearing-impaired population. *Journal of Speech, Language, and Hearing Research*, 31(3):307–316.

Thomas Schmidt and Kai Wörner. 2014. Exmaralda. In Jacques Durand, Ulrike Gut, and Gjert Kristoffersen, editors, *The Oxford handbook of corpus phonology*. Oxford University Press.

Thomas Schorrsidt. 2011. A tei-based approach to standardising spoken language transcription. *Journal of the Text Encoding Initiative*, 1.

Arjan van Hessen, Silvia Calamai, Henk van den Heuvel, Stefania Scagliola, Norah Karrouche, Jeannine Beeken, Louise Corti, and Christoph Draxler.

2020. Speech, voice, text, and meaning: A multidisciplinary approach to interview data through the use of digital tools. In *Companion Publication of the 2020 International Conference on Multimodal Interaction*, ICMI '20 Companion, page 454–455, New York, NY, USA. Association for Computing Machinery.

An Unsupervised Approach to Extract Life-Events from Personal Narratives in the Mental Health Domain

Seyed Mahed Mousavi, Roberto Negro, Giuseppe Riccardi
Signals and Interactive Systems Lab, University of Trento, Italy
{mahed.mousavi, giuseppe.riccardi} @unitn.it

Abstract

Personal Narratives are an important source of knowledge in the mental health domain. Over an extended period of time, the psychologist learns about the patient's life-events and participants from the Personal Narratives shared during each therapy session. The acquired knowledge is then used to support the patient to reach a healthier mental state by appropriate targeted feedback during each conversation. In this work, we propose an unsupervised approach to automatically extract personal life-events and participants from the patient's narratives and represent them as a personal graph. This personal graph is then updated at each interaction with the patient. We have evaluated our proposed approach on a dataset of longitudinal Italian Personal Narratives as well as a dataset of English commonsense stories.

1 Introduction

There is a growing research and clinical interest in developing conversational agents (CA) for mental health support as Personal Healthcare Agents (PHA) (Abd-alrazaq et al., 2019; Fitzpatrick et al., 2017; Inkster et al., 2018). However, the lack of appropriate domain knowledge has resulted in the abundance of rule-based dialogue systems in the mental health domain with shallow interactions and weak user engagement (Abd-Alrazaq et al., 2021). Currently available dialogue knowledge can be adequate for consumer-oriented agents or holding a free-topic social conversation. However, it can not be used to hold a dialogue about personal life-events and emotions. Meanwhile, patients' conversations in the mental health domain have a unique and complex structure since they encompass personal feelings and situations which vary across patients and interventions.

In order to carry out a personal conversation regarding the patient's life-events, it is essential to obtain the required knowledge during each interaction with the patient and from her Personal Narratives. Personal Narratives (PN) are recollections of thoughts and emotions about life-events of the patient. These narratives are used by the psychologist to identify the issues that have activated the patient's emotional state and provide support accordingly in order to reach a healthier mental status (Tammewar et al., 2019; Vromans and Schweitzer, 2011).

In this work, we present an unsupervised approach, inspired by (Chambers and Jurafsky, 2008), to automatically extract the life-events and their participants from the patient's PNs, and construct a Personal Space Graph. Figure 1 represents the work flow of our model. Through the interaction with the patient, each narrative is parsed and presented in terms of its predicates (the events, the edges of the graph) and their noun dependencies (the participants, the nodes of the graph). Each edge has an index based on its order of appearance in the narrative which makes it possible to reconstruct the order of occurrences among the events (for instance, the event "*litigo spesso*" is mentioned after "*parla male*"). Moreover, the events and participants mentioned in a recent narrative are considered to be more relevant for an ongoing interaction. Based on this assumption, older nodes and edges in the graph will become less relevant upon receiving a new narrative (presented by dashed lines in Figure 1). The obtained graph can be integrated with PHAs to automatically identify the life-event that is distressing the patient from his/her PNs to provide support and monitor its recurrence.

We have evaluated our approach on a dataset

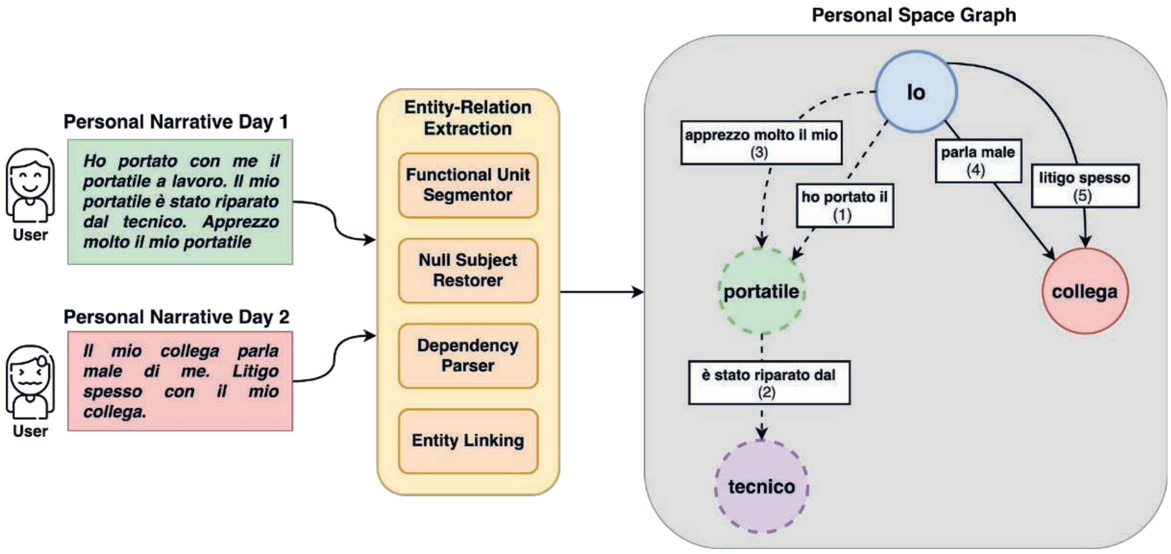


Figure 1: Each Personal Narrative (PN) is presented as a graph of patient's personal space of participants and events. Each edge and the adjacent nodes stand for an event and its corresponding participants in the narrative, respectively. Each edge (event), has an index based on its appearances in the PN which makes it possible to reconstruct the order of occurrences among the events. Moreover, events and participants extracted from prior narratives are considered less relevant for an ongoing interaction and have a lower importance score, presented by dashed lines.

of longitudinal Italian PNs collected from patients who were receiving Cognitive Behavioural Therapy to manage their distress[i]. Besides, the English adaptation of our model was evaluated in the "Story Cloze Test" setting introduced by (Mostafazadeh et al., 2016). The results show that the proposed approach obtains similar results to other unsupervised models on the English dataset, and can be a strong baseline for personal space representation and response selection in Italian.

2 Literature Review

Unsupervised Event Extraction There have been several interesting works regarding the unsupervised extraction of events and their participants from unstructured text. (Chambers and Jurafsky, 2008) introduced the concept of "Narrative Event Chain". In this work, the events with a shared participant are assumed to be parts of a uniform story. They present the events in the narrative by the verbs that have a shared participant, and the participant's role for each verb. (Chambers and Jurafsky, 2009) then extended this model to "Narrative Schema", obtained by merging different event chains extracted from one narrative into an integrated uniform schema in order to model the document by all participants across the verbs. Recently, (Hatzel and Biemann, 2021) proposed to further extend the "Narrative Schema" concept to support long documents in German language by 1) performing language adaptation of the model; and 2) dividing the event sequence into multiple strongly-connected schema in order to present different scenes in a long story.

Evaluation Criteria Regarding the evaluation of the models, the mentioned unsupervised approaches were evaluated in "Narrative Cloze Test" setting. In this setting, an event is removed from a sequence of events in a document and the task is to predict the most probable candidate for the missing event (Chambers and Jurafsky, 2008). Later however, (Mostafazadeh et al., 2016) introduced "Story Cloze Test" evaluation criterion. In this setting, the system selects a complete lexicalized sentence as the closure to a story rather than predicting the missing event. For this purpose, the authors crowd-sourced a dataset of commonsense stories, called ROCStories, with right and wrong ending sentences for each story.

3 Personal Space Graph Representation

In this work, we propose an unsupervised Entity-Relation Extraction (ERE) approach to obtain the

[i]This data collection has been approved by the Ethical Committee of the University of Trento

personal graph of life-events and participants from the user's PNs in the mental health domain. Figure 1 shows the workflow of our approach, consisting of five main modules.

Functional Unit Segmentor Upon receiving a narrative, it is first segmented into its functional units. A functional unit is a contiguous span within a message which has a coherent communicative intention (Oltean et al., 2017). The segmentation into Functional Units was performed by a seq2seq model (Zhao and Kawahara, 2019), trained to jointly perform Functional Unit segmentation and Dialogue Act (DA) tagging, based on ISO standard DA tagging in Italian (Roccabruna et al., 2020). The model was trained on the corpus of Italian dialogues in the mental health (Mousavi et al., 2021). The predictions of the model were then post-edited and adjusted by two human annotators with strong inter-annotator agreement (0.87) measured by Cohen's κ coefficient (Fournier and Inkpen, 2012).

Dependency Parser Each functional unit is then passed to the dependency parser to obtain the corresponding dependency tree, for which spaCy natural language processing library[ii] was used. Using the obtained tree and part-of-speech tags, tokens tagged as nouns and proper nouns are extracted as nodes in the graph (nominal modifier nouns are excluded in this process since they are describing/specifying characteristics of another noun). In cases that pronouns are subjects or objects of a verb, they are extracted as nodes as well.

Entity Linking In order to make sure repeated nouns or variations of the same noun are mapped to the correct node in the graph, an Entity Linking module is defined. This module queries BabelNet[iii] and ConceptNet[iv] semantic networks for the root form of the extracted nouns and matches them consequently to obtain a set of entities and participants in the narrative.

Null Subject Restorer All the verbs contained in the functional unit are extracted and controlled for possible null subject case. Null subjects are non overtly expressed subject pronouns commonly used in pro-drop languages such as Italian and Spanish (Russo et al., 2012). In this case, the subject of the verb is restored as a pronoun based on its conjugation using an out-of-the-shelf library MLCONJUG3[v] to make sure each event participant is detected and extracted correctly.

Entity-Relation Extraction Lastly, the model navigates through the dependency tree to find the verbs that connect the extracted entities as subjects and objects/oblique nominals. In cases of entity conjunctions, the same verb spans over all the entities in the same conjunction. For a better visualization, the neighbours of the verb in the dependency tree are explored to obtain an entire predicate composed by adverbs, ad-positions and auxiliaries as the edge of the graph.

The obtained graph is specific to each patient and spans over the life-events shared in the narratives. In each graph, the patient is presented as the node "*Io*" and all the other participants in the patient's PNs are connected to it by the corresponding predicate. PNs in the mental health domain are about the events that activated the patient's emotional state. Therefore, it is important to maintain the consecutive order among events in each PN as well as among subsequent PNs through several interactions with the patient. For this purpose, each edge is indexed based on its sequence of appearance in the narrative in order to reconstruct the ordered chain of events. Moreover, events extracted from prior narratives are considered less relevant to the patient's mental status, unless they re-appear in recent narratives. Therefore, these events receive lower importance score in time based on the assumption that the issue is resolved and the patient does not feel the need to re-mention it.

4 Evaluations

We have evaluated our proposed approach in two different settings in the mental health domain for Italian language. Furthermore, we have compared the performance of its English adaptation with other models in the "Story Cloze Test" setting introduced by (Mostafazadeh et al., 2016).

4.1 Personal Narratives Evaluation

We first collected a dataset PNs from Italian patients who were receiving Cognitive Behavioural Therapy to better manage their distress[vi]. Using the approach introduced priorly by (Mousavi et al., 2021), the patients were asked to write PNs

[ii] spaCy `spacy.io`
[iii] BabelNet `babelnet.org`
[iv] ConceptNet `conceptnet.io`

[v] MLCONJUG3 `pypi.org/project/mlconjug3`
[vi] This data collection has been approved by the Ethical Committee of the University of Trento

Closure Selection (Pool of 2 & 5)					Narrative Selection (Pool of 2)				
Recall	**Rand.**	**TF-IDF**	**Nouns**	**ERE**	**History Size**	**Rand.**	**TF-IDF**	**Nouns**	**ERE**
R@1 in 2	50%	71.1%	41.3%	59.0%	**2 Personal Narratives**	50%	74.4%	68.8%	71.4%
R@1 in 5	20%	51.6%	34.8%	42.7%	**5 Personal Narratives**	50%	75.3%	68.8%	72.0%

Table 1: The results of evaluating our model for Entity Relation Extraction in Italian (ERE) in two different selection settings at closure level and narrative level on a dataset of Personal Narratives collected from patients in the mental health domain.

	NC-AP	**NC-ROC**	**Nouns**	**ERE**
R@1 in 2	48.7	49.4	45.1	45.6

Table 2: The result of evaluating the English adaptation of our model in "Story Cloze Test" setting, compared with other unsupervised approaches (Mostafazadeh et al., 2016). NC-AP and NC-ROC models stand for the standard Narrative Event Chain model (Chambers and Jurafsky, 2008), with the point-wise mutual information function train on Associate Press (AP) portion of the English Gigaword Corpus and the ROCStories, respectively.

about real-life situations and events that have activated their emotional state for the period of three months. As the result, we collected 241 PNs from 18 patients with average length of 128.2 tokens per PN and average number of 11.9 PNs per patient.

Using the obtained dataset of PNs, in the first setting we evaluated the model for the task of Closure Selection. That is, the model was tasked to select the correct closure sentence for an incomplete narrative based on the participants and events (verbs) it consists of. Similar to a response-selection setting, we assessed the performance of the model using two pools of 2 and 5 candidates, each consisting of 1 correct closure and *n-1* distractors.

In the second setting, we evaluated whether the obtained graph can correctly represent a personal space of events and participants that varies for each user. To this end, the model was first presented with a set of 2 or 5 consecutive PNs from a specific patient as history. Once the corresponding personal space graph was extracted, the model was tasked to select the next possible PN from that patient from a pool of 2 candidates, consisting of the correct PN and a distractor (a PN written by a different user.)

The results of these evaluations are presented in Table 1. In the first scenario, while TF-IDF manages to be a strong baseline, our proposed system outperforms the Random baseline and has a much higher success rate than the selection solely based on the recurrence of the nouns. Moreover, by raising the task difficulty and increasing the pool size to 5, our model maintains the same performance trend. Regarding the second evaluation, the results indicate that the recurrence of the nouns is an important factor for the model to select the next possible PN. Nevertheless, our model manages to outperform this baseline by considering the predicates as an additional factor, and get closer to TF-IDF scores.

4.2 Story Cloze Test

In order to compare the performance of our model with other unsupervised approaches, the English adaption of the model was evaluated in the "Story Cloze Test" setting. In this setting, the model is tasked to select the most probable ending for a four-sentence story from a pool of 2, consisting of the right ending and the wrong one. (Mostafazadeh et al., 2016). The result of this evaluation for the test set of 3744 stories is presented in Table 2, indicating that our model performance is inline with other unsupervised approaches.

5 Conclusion

In this work, we present an approach to automatically extract life-events and participants from patients' Personal Narratives in the mental health domain and represent them as a personal graph. This graph can be a source of knowledge for Personal Healthcare Agents (PHA) in this domain, to automatically identify the life-event that is activating the user's emotional state and causing distress.

We evaluated our model on a domain-specific dataset of Personal Narratives in Italian as well as an open-domain dataset of commonsense stories in English. The results indicate that our proposed model performs in-line with other unsupervised alternatives and can be a strong baseline for automatic extraction of life-events from Personal Narratives in Italian.

Acknowledgments

The research leading to these results has received funding from the European Union – H2020 Programme under grant agreement 826266: COADAPT.

References

Alaa A Abd-alrazaq, Mohannad Alajlani, Ali Abdallah Alalwan, Bridgette M Bewick, Peter Gardner, and Mowafa Househ. 2019. An overview of the features of chatbots in mental health: A scoping review. *International Journal of Medical Informatics*, 132:103978.

Alaa A Abd-Alrazaq, Mohannad Alajlani, Nashva Ali, Kerstin Denecke, Bridgette M Bewick, and Mowafa Househ. 2021. Perceptions and opinions of patients about mental health chatbots: Scoping review. *Journal of Medical Internet Research*, 23(1):e17828.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised learning of narrative event chains. In *Proceedings of ACL-08: HLT*, pages 789–797.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 602–610.

Kathleen Kara Fitzpatrick, Alison Darcy, and Molly Vierhile. 2017. Delivering cognitive behavior therapy to young adults with symptoms of depression and anxiety using a fully automated conversational agent (woebot): a randomized controlled trial. *JMIR mental health*, 4(2):e19.

Chris Fournier and Diana Inkpen. 2012. Segmentation similarity and agreement. *arXiv preprint arXiv:1204.2847*.

Hans Ole Hatzel and Chris Biemann. 2021. Towards layered events and schema representations in long documents. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 32–39, Online, June. Association for Computational Linguistics.

Becky Inkster, Shubhankar Sarda, and Vinod Subramanian. 2018. An empathy-driven, conversational artificial intelligence agent (wysa) for digital mental well-being: real-world data evaluation mixed-methods study. *JMIR mHealth and uHealth*, 6(11):e12106.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849.

Seyed Mahed Mousavi, Alessandra Cervone, Morena Danieli, and Giuseppe Riccardi. 2021. Would you like to tell me more? generating a corpus of psychotherapy dialogues. In *Proceedings of the Second Workshop on Natural Language Processing for Medical Conversations*, pages 1–9, Online, June. Association for Computational Linguistics.

Horea-Radu Oltean, Philip Hyland, Frédérique Vallières, and Daniel Ovidiu David. 2017. An empirical assessment of rebt models of psychopathology and psychological health in the prediction of anxiety and depression symptoms. *Behavioural and cognitive psychotherapy*, 45(6):600–615.

Gabriel Roccabruna, Alessandra Cervone, and Giuseppe Riccardi. 2020. Multifunctional iso standard dialogue act tagging in italian. *Seventh Italian Conference on Computational Linguistics (CLiC-it)*.

Lorenza Russo, Sharid Loáiciga, and Asheesh Gulati. 2012. Improving machine translation of null subjects in italian and spanish. In *Proceedings of the Student Research Workshop at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 81–89.

Aniruddha Tammewar, Alessandra Cervone, Eva-Maria Messner, and Giuseppe Riccardi. 2019. Modeling user context for valence prediction from narratives. In *Interspeech*, page 3252–3256.

Lynette P Vromans and Robert D Schweitzer. 2011. Narrative therapy for adults with major depressive disorder: Improved symptom and interpersonal outcomes. *Psychotherapy research*, 21(1):4–15.

Tianyu Zhao and Tatsuya Kawahara. 2019. Joint dialog act segmentation and recognition in human conversations using attention to dialog context. *Computer Speech & Language*, 57:108–127.

An Italian Question Answering System Based on Grammars Automatically Generated from Ontology Lexica

Gennaro Nolano[1], **Mohammad Fazleh Elahi**[2], **Maria Pia di Buono**[1], **Basil Ell**[2,3] **and Philipp Cimiano**[2]
1. UniOr NLP Research Group, University of Naples "L'Orientale", Italy
2. Cognitive Interaction Technology Center, Bielefeld University, Germany
3. Department of Informatics, University of Oslo
gnolano,mpdibuono@unior.it,
{melahi,bell,cimiano}@techfak.uni-bielefeld.de

Abstract

The paper presents an Italian question answering system over linked data. We use a model-based approach to question answering based on an ontology lexicon in lemon format. The system exploits an automatically generated lexicalized grammar that can then be used to interpret and transform questions into SPARQL queries. We apply the approach for the Italian language and implement a question answering system that can answer more than 1.6 million questions over the DBpedia knowledge graph.

1 Introduction

As the amount of linked data published on the Web keeps increasing, there is an expanding demand for multilingual tools and user interfaces that simplify the access and browsing of data by end-users, so that information can be explored in an intuitive way. This need is what motivated the development of tools such as Question Answering (QA) systems, whose main aim is to make users be able to explore complex datasets and an ever growing amount of data in an intuitive way, through natural language.

While the default approach for many NLP tasks has recently been represented by machine learning systems, the use of such approaches (Chakraborty et al., 2019) for QA over RDF data suffers from lack of controllability, making the governance and incremental improvement of the system challenging, not to mention the initial effort of collecting and providing training data for a specific language.

An alternative is the so-called model-based approach to QA, in which a model is first used to specify how concepts and relations are realized in natural language, and then this specification is employed to interpret questions from users. One such system is the one proposed by (Benz et al., 2020), which makes use of a lexicon in lemon format (McCrae et al., 2011) to specify how the vocabulary elements of an ontology or knowledge graph (e.g., entities and relations from a Knowledge Graph) are realized in natural language.

The previous work on this approach shows how, leveraging on lemon lexica, question answering grammars can be automatically generated, and those can, in turn, be used to interpret questions and then parse them into SPARQL queries. A QA web application developed in previous work (Elahi et al., 2021) has further shown that such QA systems can scale to large numbers of questions and that the performance of the system is practically real-time from an end-user perspective.

In this work we describe the extension to the Italian language of the model-based approach described in (Benz et al., 2020) and the QA system described in (Elahi et al., 2021). By doing so, we develop a QA system that can answer more than 1.6 million Italian questions over the DBpedia knowledge graph[1].

2 Related Work

Besides the goal of creating QA systems that are robust and have high performance, an important goal is also to develop systems that can be ported to languages other than English. The interest in other languages is, for example, explicitly stated in the Multiple Language Question Answering Track at CLEF 2003 (Magnini et al., 2004), that includes Italian among others.

One of the earlier attempts in this regard has been the DIOGENE model (Magnini et al., 2002; Tanev et al., 2004), which exploits linguistic tem-

[1] https://www.dbpedia.org/

plates and keyword recognition to answer questions over document collections. Other efforts have been made in the QALL-ME project (Cabrio et al., 2007; Cabrio et al., 2008; Óscar Ferrández et al., 2011), where a system was created for the tourism domain through an instance-based method, that is by clustering together similar question-answer pairs.

More recently, the QuASIt model (Pipitone et al., 2016), makes use of the *Construction Grammar* and an abstraction of cognitive processes to account for the inherent fluidity of language, while exploiting linguistic and domain knowledge (in the form of an ontology) to answer essay and multiple choice questions. Similarly, the authors of (Leoni et al., 2020) built a system to answer questions regarding a specific domain using IBM Watson services and online articles as source of information.

These kind of systems, built to answer questions using textual information, have been largely growing in recent years, especially since the availability of large QA datasets such as the Stanford Question Answering Dataset (SQuAD)[2], which allows to train complex deep learning models with millions of parameters (Rajpurkar et al., 2016; Rajpurkar et al., 2018). While the performance shown by these models is impressive, they suffer from major drawbacks: first of all, they need an extremely large dataset to be trained on, making the porting of such a system to another language extremely demanding;[3] furthermore, they show a lack of controllability in the sense that it is unclear which new examples are to be added to make a new question answerable. This makes systems opaque and difficult to maintain.

The MULIB system (Siciliani et al., 2019) tackles the problem of answering questions in Italian over structured data. The system is based on a modified version of the automaton developed for CANaLI (Mazzeo and Zaniolo, 2016), but it employs a Word2Vec model (Mikolov et al., 2013) to allow for more flexibility in language use. In contrast to these trained approaches, we present a model that generates (i) a deeper interconnection of semantic and syntactic information through the integration of a lemon lexicon with the DBpedia ontology, and (ii) the focus on Linked Open Data as a source of knowledge.

3 Methodology

The architecture consists of two components: (i) the grammar generator and (ii) the QA component. The approach to grammar generation for different syntactic frames according to LexInfo (Cimiano et al., 2011) for the English language was described in a previous work (Benz et al., 2020). In this paper we show that, through a simple language adaptation, we are able to adjust the system so that the system also accepts questions in Italian language.

In a nutshell, the grammar generation approach relies on a mapping between syntactic constructions and classes and properties from a given ontology and/or knowledge graph. This generation process makes use of several *frames*, each describing the linguistic realizations of specific properties that might appear in questions. Thus, the frames employed in this work are: *NounPPFrame*, *TransitiveFrame*, *IntransitivePPFrame*, *AdjectiveAttributive* and *AdjectiveGradable*.

For example, the (lexicalized) construction for the NounPPFrame *'the capital of X'*, can be regarded as expressing the DBpedia property `dbo:capital`, with Country as domain and City as range. This would lead to the generation of the following questions:

- What is the **capital** of X (Country)?
- Which city is the **capital** of X (Country)?

Similar grammar generation rules exist for transitive constructions (TransitiveFrame) as well as constructions involving an intransitive verb with a prepositional complement (IntransitivePPFrame) as well as adjective constructions in attributive (AdjectiveAttributive) and predicate form (AdjectiveGradable).

In the context of this work, we adapted the generation of rules to the Italian language, without extending or modifying the existing types of constructions[4].

In adapting the grammar generation to Italian, we had to accommodate for the following language-specific properties:

- Sentence order, e.g., in sentence starting with interrogative pronouns the subject has to be

[2] `https://rajpurkar.github.io/SQuAD-explorer/`

[3] The Italian translation for SQuAD, for example, has been described in Croce et al. (2018)

[4] The code for our grammar generation for Italian is available at `https://github.com/fazleh2010/question-grammar-generator`

placed at the end of the sentence, e.g., *Dove si trova Vienna?* (Where is Vienna?)

- The presence of auxiliary verbs, either *avere* (have) or *essere* (be), in compound tenses;
- Interrogative pronoun rules, e.g., *chi* (who) is invariable and refers only to people;
- The use of interrogative adjectives, e.g., *quale* (which);
- The use of different prepositions, either simple or articulated, on the basis of range/domain semantics (e.g., toponyms might require different prepositions);
- The presence of a determiner/articulated preposition on the basis of range/domain semantics (e.g., toponyms are preceded by a determiner when the noun refers to a country).

```

:lexicon_en a lemon:Lexicon ;
  lemon:language "it" ;
  lemon:entry :capital_of ;
  lemon:entry :di .

:capital_of a lemon:LexicalEntry ;
  lexinfo:partOfSpeech lexinfo:noun ;
  lemon:canonicalForm :capital_form ;
  lemon:synBehavior :capital_of_nounpp ;
  lemon:sense :capital_sense1 .

:capital_form a lemon:Form ;
  lemon:writtenRep "capitale"@it .

:capital_of_nounpp a lexinfo:NounPPFrame ;
  lexinfo:copulativeArg :arg1 ;
  lexinfo:prepositionalAdjunct :arg2 .

:capital_sense1 a lemon:OntoMap, lemon:LexicalSense ;
  lemon:ontoMapping :capital_sense1 ;
  lemon:reference dbo:capital ;
  lemon:subjOfProp :arg2 ;
  lemon:objOfProp :arg1 ;
  lemon:condition :capital_condition .

:capital_condition a lemon:condition ;
  lemon:propertyDomain dbo:Country ;
  lemon:propertyRange dbo:City .

:arg2 lemon:marker :di .

:di a lemon:SynRoleMarker ;
  lemon:canonicalForm [ lemon:writtenRep "della"@it ] ;
  lexinfo:partOfSpeech lexinfo:preposition .
```

Figure 1: Lemon entry for the relational noun *'capitale della'*

Consider the lemon lexical entry in Figure 1[5] for the relational noun *'capitale della'*. The entry states that the canonical written form of the entry is *"capitale"*. It states that the entry has a NounPPFrame as syntactic behaviour, that is it corresponds to a copulative construction *X è la capitale della Y* with two arguments, where `copulativeArg` corresponds to the copula subject X and the *prepositional adjunct* corresponds to the prepositional object Y.

We give examples for the different syntactic frames below to illustrate the behaviour of the Italian grammar generation.

NounPPFrame Assuming that in the corresponding lemon lexicon we model the connection between the NounPP construction *capitale della* (capital of) as referring to the property `dbo:capital` with domain Country and range City, we can generate questions automatically such as:

1. *Qual è la capitale della* (What is the capital of) (X—Country_NP)?
2. *Quale città è la capitale della* (Which city is the capital of) (X—Country_NP)?

where *X* is a placeholder allowing to fill in a particular country, e.g. *Germania* (Germany), or a noun phrase, e.g., *paese dove si parla tedesco* (the country where German is spoken).

TransitiveFrame Assuming that the lemon lexicon captures the meaning of the construction X *'scrive'* (write) Y as referring to the property `dbp:author`, with Song as domain and Person as range, the following questions would then be covered by an automatically generated grammar:

1. *Chi ha scritto* (Who wrote) (X—Song_NP)?
2. *Quale cantante ha scritto* (Which singer wrote) (X—Song_NP)?
3. *Quale* (Which) (X—Song_NP) *è stata scritta da* (was written by) (Y—Person_NP)?

IntransitivePPFrame Assuming that the lemon lexicon captures the meaning of the construction *'X pubblicare nel Y'* (*'X published in Y'*) as representation of the property `dbp:published`, with Song as its domain and Date as its range, the following questions would be generated:

1. *Quando è stata pubblicata (X—Song_NP)?* (When was (X—Song_NP) published?),
2. *Quale (X—Song_NP) è stata pubblicata nel (Y—date)?* (Which (X—Song_NP) was published in (Y—date)?
3. *In quale data è stata pubblicata* (In which date was) (X—Song_NP)?

[5]In this paper we abbreviate URIs with the namespace prefixes `dbo`, `dbp`, `lemon`, and `lexinfo` which can be expanded into `http://dbpedia.org/ontology/`, `http://dbpedia.org/property/`, `https://lemon-model.net/lemon#`, and `http://www.lexinfo.net/ontology/2.0/lexinfo#`, respectively.

LexInfo Frame	Syntactic Pattern	Question Sample
NounPP	WDT/WP V* DT [noun] IN DT [domain] WDT `dbo:range` V* DT [noun] IN [domain]? WDT/WP V* DT [noun] in [domain] [range] V* DT [noun] IN (DT) [domain]	*Qual è la* ***capitale della*** *Germania?* *Quale città è la* ***capitale della*** *Germania?* *Chi era la* ***moglie di*** *Abraham Lincoln?* *Rita Wilson è la* ***moglie di*** *Tom Hanks?*
AdjectiveAttributive	WDT V* DT `dbo:range` [adjective] [domain] VB (DT) [adjective]	*Chi era un vescovo cristiano* ***spagnolo****?* *Barack Obama è un* ***democratico****?*
AdjectiveGradable	WRB V* [adjective] DT [domain] WDT V* DT [domain] JJS IN (DT) [range]	*Quanto è* ***lungo il Barguzin?*** *Qual è la montagna* ***più alta*** *della Germania?*
Transitive	WP V* [domain] WDT `dbo:range` V* [domain] WP V* DT [domain] WDT `dbo:range` V* DT [domain] [domain] V* [range]	*Chi* ***ha scritto*** *Ziggy Stardust?* *Quale cantante* ***ha scritto*** *Ziggy Stardust?* *Chi* ***ha fondato*** *C&A?* *Quale persona* ***ha fondato*** *C&A?* *Socrate* ***ha influenzato*** *Aristotele?*
IntransitivePP	WRB VB [domain] IN WDT `dbo:domain` VB [range] WDT `dbo:domain` VB IN [range] [domain] V* IN [range]	*Quando* ***è iniziata*** *l'operazione Overlord?* *In quale data* ***è iniziata*** *l'operazione Overlord?* *Quale libro* ***è stato pubblicato*** *nel 1563?* *Il libro dei martiri di Foxe* ***è stato pubblicato*** *nel 1563?*

Table 1: Italian Patterns and Questions

Frame type	#Entries	#Grammar rules	#Questions
NounPPFrame	113	226	1,010,234
TransitiveFrame	41	124	595,854
IntransitivePPFrame	58	116	52,040
AdjectiveAttributiveFrame	29	130	10,025
AdjectiveGradable	8	24	3,123
Total	**249**	**620**	**1,671,276**

Table 2: Frequencies of entries with a certain frame type. The entries are created manually; the rules and questions are generated automatically.

AdjectiveAttributive and AdjectiveGradable Assuming that the lemon lexicon would capture the meaning of the (gradable) adjective *lungo* (long) as referring to the ontological property `dpb:length`, the grammar generation approach would generate the following types of questions:

1. *Quanto è lungo il* (How long is the) (X—River_NP)?
2. *Qual è il fiume più lungo (del mondo, del Kentucky)?* (What is the longest river in (*the world*, *Kentucky*)?).

The rules implemented for the generation of Italian questions are shown in further detail in Table 1. In particular, we use the tagset[6] from the Penn Treebank Project (Marcus et al., 1993), with `V*` defining all possible forms of a given verb, words in brackets defining nouns/verbs/adjectives that realize a specific property, and `dbo:range`/`dbo:domain` defining the possible labels that may represent classes (e.g., `dbo:Country` might be represented by either *paese* or *stato*).

4 Results

We apply our system to the DBpedia dataset and manually created a lemon lexicon comprising of 249 lexical entries[7]. Table 2 shows the number of grammar rules and questions generated for each syntactic type. Altogether, the approach generates 620 grammar rules and about 1.6 million questions. The web-based demonstration is available online[8].

We used the training set of multilingual QALD-

[6] `https://www.sketchengine.eu/english-treetagger-pipeline-2/`

[7] `https://scdemo.techfak.uni-bielefeld.de/quegg-resources/`

[8] `https://webtentacle1.techfak.uni-bielefeld.de/quegg/`

7[9] to evaluate our approach. QALD-7 contains a total of 214 questions over linked data, covering for more relations than the ones we considered so far. In order to overcome this issue, a total of 109 entries were added to our system (22 NounPPFrame, 41 TransitiveFrame, 41 IntransitiveFrame, 1 AdjectiveAttributiveFrame and 4 AdjectiveGradable).

Precision	0.485
Recall	0.224
F-Measure	**0.307**

Table 3: Evaluation results against QALD-7

The results of the evaluation process (Table 3) show a quite satisfying precision, but a low recall. The main reason behind such results is related to the presence of different types of questions in QALD. Indeed, besides single-triple questions, QALD presents also complex questions referring to more than one triple, e.g., *A quale movimento artistico apparteneva il pittore de I tre ballerini?* (What was the artistic movement of the author of The Three Dancers?), which are not covered yet by our model. Nevertheless, when taking into account all the questions in QALD-7, our system recognizes 46.98% (101 questions) of the total set of questions.

5 Conclusion and Future Work

We presented an approach to developing Italian QA systems over linked data that relies on the automatic generation of grammars from corresponding lemon lexica describing how elements of the dataset are realized in natural language. The approach is controllable, since the introduction of a lexical entry increases the question coverage in a fully predictable way. Our proof-of-concept implementation over DBpedia covers 1.6 million questions generated from 249 lemon entries.

In future work, we intend to further automatize grammar generation by using LexExMachina (Ell et al., 2021), which induces lexicon entries bridging the gap between ontology and natural language from a corpus in an unsupervised manner.

Acknowledgments This work has been funded by the European Commission under grant 825182 (Prêt-à-LLOD) as well as Nexus Linguarum Cost Action. M.P. di Buono has been partially supported by Programma Operativo Nazionale Ricerca e Innovazione 2014-2020 - Fondo Sociale Europeo, Azione I.2 “Attrazione e Mobilità Internazionale dei Ricercatori” Avviso D.D. n 407 del 27/02/2018. B. Ell has been partially supported by the SIRIUS centre: Norwegian Research Council project No 237898.

[9] `https://github.com/ag-sc/QALD`

References

Viktoria Benz, Philipp Cimiano, Mohammad Fazleh Elahi, and Basil Ell. 2020. Generating Grammars from lemon lexica for Questions Answering over Linked Data: a Preliminary Analysis. In *NLIWOD workshop at ISWC*, volume 2722, pages 40–55.

Elena Cabrio, Bonaventura Coppola, Roberto Gretter, Milen Kouylekov, Bernardo Magnini, and Matteo Negri. 2007. Question answering based annotation for a corpus of spoken requests. In *Proceedings of the workshop on the Semantic Representation of Spoken Language*, volume 31.

Elena Cabrio, Milen Kouylekov, Bernardo Magnini, Matteo Negri, Laura Hasler, Constantin Orasan, David Tomás, Jose Luis Vicedo, Guenter Neumann, and Corinna Weber. 2008. The QALL-ME benchmark: a multilingual resource of annotated spoken requests for question answering. In *LREC'08*.

Nilesh Chakraborty, Denis Lukovnikov, Gaurav Maheshwari, Priyansh Trivedi, Jens Lehmann, and Asja Fischer. 2019. Introduction to Neural Network based Approaches for Question Answering over Knowledge Graphs. *CoRR*, abs/1907.09361.

Philipp Cimiano, Paul Buitelaar, John P. McCrae, and Michael Sintek. 2011. LexInfo: A declarative model for the lexicon-ontology interface. *JWS*, 9(1):29–51.

Danilo Croce, Alexandra Zelenanska, and Roberto Basili. 2018. Neural learning for question answering in italian. In *AI*IA 2018*, pages 389–402.

Mohammad Fazleh Elahi, Basil Ell, Frank Grimm, and Philipp Cimiano. 2021. Question Answering on RDF Data based on Grammars Automatically Generated from Lemon Models. In *SEMANTiCS Conference, Posters and Demonstrations*.

Basil Ell, Mohammad Fazleh Elahi, and Philipp Cimiano. 2021. Bridging the Gap Between Ontology and Lexicon via Class-Specific Association Rules Mined from a Loosely-Parallel Text-Data Corpus. In *LDK 2021*, pages 33:1–33:21.

Chiara Leoni, Ilaria Torre, and Gianni Vercelli. 2020. ConversIAmo: Improving Italian Question Answering Exploiting IBM Watson Services. In *Text, Speech, and Dialogue*, pages 504–512.

Bernardo Magnini, Matteo Negri, Roberto Prevete, and Hristo Tanev. 2002. Mining Knowledge from Repeated Co-Occurrences: DIOGENE at TREC 2002.

Bernardo Magnini, Simone Romagnoli, Alessandro Vallin, Jesús Herrera, Anselmo Peñas, Víctor Peinado, Felisa Verdejo, and Maarten de Rijke. 2004. The Multiple Language Question Answering Track at CLEF 2003. In *Comparative Evaluation of Multilingual Information Access Systems*, pages 471–486.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Comput. Linguist.*, 19(2):313—-330.

Giuseppe M. Mazzeo and Carlo Zaniolo. 2016. Answering controlled natural language questions on RDF knowledge bases. In *EDBT*, pages 608–611.

John P. McCrae, Dennis Spohr, and Philipp Cimiano. 2011. Linking Lexical Resources and Ontologies on the Semantic Web with Lemon. In *ESWC Conference*, pages 245–259.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26.

Arianna Pipitone, Giuseppe Tirone, and Roberto Pirrone. 2016. QuASIt: A Cognitive Inspired Approach to Question Answering for the Italian Language. volume 10037, pages 464–476.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. *CoRR*, abs/1606.05250.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You Don't Know: Unanswerable Questions for SQuAD. *CoRR*, abs/1806.03822.

Lucia Siciliani, Pierpaolo Basile, Giovanni Semeraro, and Matteo Mennitti. 2019. An italian question answering system for structured data based on controlled natural languages. In *CLiC-it*.

Hristo Tanev, Matteo Negri, Bernardo Magnini, and Milen Kouylekov. 2004. The DIOGENE question answering system at CLEF-2004. volume 3491, pages 435–445.

Óscar Ferrández, Christian Spurk, Milen Kouylekov, Iustin Dornescu, Sergio Ferrández, Matteo Negri, Rubén Izquierdo, David Tomás, Constantin Orasan, Guenter Neumann, Bernardo Magnini, and Jose Luis Vicedo. 2011. The QALL-ME Framework: A specifiable-domain multilingual Question Answering architecture. *Journal of Web Semantics*, 9(2):137–145.

Tackling Italian University Assessment Tests with Transformer-Based Language Models

Daniele Puccinelli[1], **Silvia Demartini**[1], **Pier Luigi Ferrari**[2]
1. University of Applied Sciences and Arts of Southern Switzerland, Switzerland
2. University of Eastern Piedmont, Italy
{daniele.puccinelli, silvia.demartini}@supsi.ch,
pierluigi.ferrari@uniupo.it

Abstract

Cloze tests are a great tool to asses reading proficiency as well as analytical thinking, and are therefore employed in admission and assessment tests at various levels of the education system in multiple countries. In Italy, cloze tests are administered to incoming university students to ascertain their starting level. The goal of a cloze test is to determine several tokens that have been pre-deleted from a text; this is largely equivalent to the well-known NLP task of missing token prediction. In this paper, we show that cloze tests can be solved reasonably well with various Transformer-based pre-trained language models, whose performance often compares favorably to the one of incoming Italian university students.

1 Introduction

A cloze test is a reading comprehension assessment where participants are presented with a text in which selected tokens have been replaced with blanks. The goal is for the participant to choose tokens (often from a list) and use them to replace the blanks based on the overall context. Typically, one every 5-10 tokens is replaced with a blank.

Cloze tests are one of the most common linguistic tests in use for formative and summative purposes, along with written responses, multiple-choice tests, matching tests, ordering tests, summarizing tests etc. (Lugarini, 2010). Cloze tests were originally introduced in the United States in the 1950s to measure the readability of texts (Taylor, 1953) and involved the random and not predetermined deletion of words that appeared at predefined intervals. This method was too general for didactic and evaluation purposes, but it was quickly adapted and became very widespread as a teaching and testing technique (Radice, 1978). In education, cloze tests have become more targeted: words are deleted according to various criteria, depending on the specific testing goals. In general, cloze tests are designed to evaluate one of the following:

- field-specific knowledge acquisition, by asking to insert appropriate words about a topic or a discipline;
- text comprehension, by asking for information that can be inferred from the text (with no prior domain knowledge);
- linguistic aspects, typically with respect to L1, L2 and FL (foreign language) acquisition at different levels (i. e. vocabulary, specific parts of speech etc.).

If carefully designed, cloze tests can be a very effective tool at all educational levels; on the other hand, cloze tests may also show some limits and issues in assessing linguistic competence (Chiari, 2002), as they necessarily offer a partial and contextual view. However, the long tradition of study and use in the fields of educational linguistics and linguistic makes it very interesting to compare human and automatic performances in dealing with cloze tests.

2 Methodology

We tackle the cloze tests in our dataset with pre-trained language models based on the Transformer architecture (Vaswani et al., 2017). We employ both autoencoding and autoregressive models. Given the very small number of datapoints at our disposal, model fine-tuning is not a viable option; therefore, we use pre-trained versions of such models, all of which are publicly available through Huggingface at the time of writing (summer 2021).

Dataset. Our dataset contains eleven cloze tests focusing on general linguistic competence that were administered to incoming first year students at the University of Eastern Piedmont in the cities of Alessandria and Vercelli in northwestern Italy between 2017 and 2019. Each cloze test was taken by a number of students in the low three digits, ranging from 130 to 390. As these are university-level tests, all students had at least a high school diploma. Most of the students were L1. The tests were offered on-site (in information technology classrooms) through the Moodle Learning Platform.

Our dataset contains two types of cloze tests: nine restricted tests where a list of options is provided for each blank to be filled, and two unrestricted tests where a global list of options is provided for all blanks with no token subgrouping (i.e., with no information about which tokens are supposed to go where). In the two unrestricted tests and three of the nine restricted ones, the list(s) contain single token options. In the other six restricted tests, the lists contain at least one multiple token option (e.g., *il quale* or *con l'utilizzo*). These cloze tests involved both function words as well as content words with both lexical and grammatical meanings

Autoencoding models. Our choices for autoencoding models are BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), DistilBERT (Sanh et al., 2019), and ELECTRA (Clark et al., 2020).

BERT is a natural choice because one of its two pre-training tasks is masked language modeling: a fraction of tokens in the pre-training data are masked so that BERT can be pre-trained to reconstruct them. Viewed as an NLP task, a cloze test is a special case of masked language modeling task where tokens are masked in an adversarial fashion: instead of choosing tokens to be masked uniformly at random, tokens are masked to challenge the test taker to reconstruct the meaning of the original text. Because a cloze test is functionally equivalent to a masked language modeling task, it is reasonable to use pre-trained BERT with no further task-specific fine-tuning.

RoBERTa improves on the original BERT by focusing on the aforementioned masked language modeling task and removing the other pre-training task (next sentence prediction). UmBERTo[1] is a RoBERTa-based model that contains some interesting optimization such as SentencePiece and Whole Word Masking. UmBERTo has been shown to perform very well compared to other BERT-based models (Tamburini, 2020).

DistilBERT (Sanh et al., 2019) is a more compact language model pre-trained with knowledge distillation(Hinton et al., 2015), a technique that uses the output of a larger teacher network to train a smaller student network. BERTino (Muffo and Bertino, 2020) is an Italian DistilBERT model that was recently proposed as a lightweight alternative to BERT specifically for the Italian language.

ELECTRA is pre-trained with replaced token detection: instead of being masked, tokens are replaced with plausible alternatives sampled from a generator network; the model is then pre-trained to discriminate whether each token was replaced by a generator sample or not. At the outset of this study, the authors posited that replaced token detection is enough to make ELECTRA reasonably ready to tackle cloze tests with no further task-specific fine-tuning; this is indeed the case, as confirmed by the results shown in Table 1.

To summarize, we employ the following autoencoding models (all cased, as the cloze tests in our dataset contain case-sensitive options):

- multilingual BERT-base[2] (BERT multi), which serves as a baseline for autoencoding models;
- the Bayerische Staatsbibliothek's Italian BERT model[3] (BERT it);
- a smaller version of multilingual BERT-base[4] (BERT it LWYN) based on the Load What You Need concept described in (Abdaoui et al., 2020);
- UmBERTo[5] as the representative of the RoBERTa family.
- BERTino[6] as the representative of the DistilBERT family;

[1]https://github.com/musixmatchresearch/UmBERTo
[2]bert-base-multilingual-cased
[3]dbmdz/bert-base-italian-xxl-cased
[4]Geotrend/bert-base-it-cased
[5]Musixmatch/UmBERTo-commoncrawl-cased-v1
[6]indigo-ai/BERTino

- the Bayerische Staatsbibliothek's Italian ELECTRA model[1].

Autoregressive models. The key limitation of masked language modeling as a proxy for cloze test is the focus on single token masking. Therefore, autoencoding models are not applicable to the six cloze tests in our dataset that feature at least one multiple token option. (In some cases, the multiple token options are consistently among the incorrect options; using our autoenconding models in such cases would therefore skew the results in the models' favor.) For these tests, we employ a simple strategy based on autoregressive models: we iterate over all possible substitutions given the options offered by a test and choose the one with the lowest perplexity as determined by each of our autoregressive language models, all of which are from the GPT-2(Radford et al., 2019) family and include the following:

- a standard GPT-2 model[2], which serves as a performance lower bound (Vanilla GPT-2);
- a *recycled* version of GPT-2[3] transferred to the Italian language(de Vries and Nissim, 2020) (Recycled GPT-2);
- GePpeTto[4](Mattei et al., 2020), the first generative language model for Italian, also built using the GPT-2 architecture.

3 Results

The results of our study are summarized in Table 1. We report the results obtained by the human test takers and the models for each of the eleven cloze tests in our dataset as well as aggregates (mean values) over the whole dataset. For each cloze test, we report the number of blanks to be filled (*Questions*, which varies from 4 to 6), the number of human test takers (*Human count*), as well as with the mean and the standard deviation of the scores. Each test is identified by the initial of its topic (S=Science, L=Legal, G=Geometry, R=Reasoning, E=Education, H=History, T=Technology) along with a numeral to disambiguate multiple tests on the same topic. As previously mentioned, two tests are unrestricted (all the provided options can go anywhere in the text) and the others are restricted (there are specific option lists for each blank to be filled). As previously explained, six tests (L2, G2, E, H1, H2, T) contain at least one multi-token option and are only tackled with autoregressive models. On average, we observe that:

- humans do better than the best model (Electra) by eight percentage points;
- Electra, UmBERTo, and GePpeTto are the top three performers;
- Vanilla GPT-2 aside, BERT it LWYN comes in last and underperforms BERT it multilingual.

Averages, however, hide the enormous gap between restricted and unrestricted tests. We illustrate this gap in Table 2, which compares these two categories of tests model by model and also shows averages across autoencoding and autoregressive models (computed over the best models for each category, i.e., without BERT-base-it LWYN and BERT-base-multi for autoencoding models and without Vanilla GPT-2 for autoregressive models). This leads us to the following observations:

- our best autoencoding models outperform the human average;
- as expected, our models perform much better in restricted tests (we see a gap of 30 percentage points for autoencoding model and 10 points for autoregressive models);
- autoregressive models outperform autoenconding models in unrestricted tests, while the converse holds in restricted tests;
- humans perform similarly on both our restricted and unrestricted tests (and so does our performance lower-bound, Vanilla GPT-2).

In our restricted tests, UmBERTo and Electra outperform the human average and emerge as the top performers among our models. Though far below the human average, GePpeTto and Recycled GPT-2 are the two top performers in unrestricted tests, where none of the autoencoding model reach the pass threshold of 0.6. Vanilla GPT-2 aside, BERT it LWYN comes in last and underperforms BERT it multilingual in restricted tests while matching its baseline performance in unrestricted tests.

[1] dbmdz/electra-base-italian-xxl-cased-generator
[2] `https://huggingface.co/gpt2`
[3] GroNLP/gpt2-medium-italian-embeddings
[4] LorenzoDeMattei/GePpeTto

	S1	L1	G1	R	S2	L2	G2	E	H1	H2	T	Ave.
Restricted	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
Questions	6	6	6	4	4	6	6	6	5	5	6	
Human count	253	300	154	130	291	253	390	300	316	157	184	
Human mean	0.77	0.88	0.82	0.84	0.96	0.89	0.96	0.88	0.9	0.55	0.87	0.85
Humans std	0.21	0.16	0.15	0.25	0.1	0.14	0.1	0.14	0.14	0.25	0.14	
UmBERTo	0.34	0.68	0.76	1	1	-	-	-	-	-	-	0.76
BERTino	0.34	0.76	0.67	0.76	1	-	-	-	-	-	-	0.71
Electra	0.34	0.84	0.67	1	1	-	-	-	-	-	-	0.77
BERT it	0.34	0.84	0.67	0.76	1	-	-	-	-	-	-	0.72
BERT it LWYN	0	0.68	0.33	0.76	0.76	-	-	-	-	-	-	0.51
BERT multi	0	0.68	0.33	1	0.76	-	-	-	-	-	-	0.55
GePpeTto	0.34	1	0.5	0.76	0.76	0.67	0.84	1	1	0.4	1	0.75
Recycled GPT-2	0.34	0.92	0.5	0.76	0.76	0.83	0.66	0.84	0.8	0.6	0.84	0.71
Vanilla GPT-2	0.16	0.5	0.17	0.5	0.5	0.5	0.5	0.16	0.4	0	0.16	0.32

Table 1: Performance of various autoencoding and autoregressive language models on 11 different Italian-language cloze tests on various topics (S=Science, L=Legal, G=Geometry, R=Reasoning, E=Education, H=History, T=Technology) and comparison to human performance (the number of students who took each of the tests in reported as *Human count* along with the sample mean and the standard deviation of the scores).

	Unres.	Res.
Humans	0.83	0.85
UmBERTo	0.51	**0.92**
BERTino	0.55	0.81
Electra	0.59	**0.89**
BERT-base-it	0.59	0.81
BERT-base-it LWYN	0.34	0.62
BERT-base-multi	0.34	0.70
Autoencoding ave.	0.56	**0.86**
GePpeTto	0.67	0.77
Recycled GPT-2	0.63	0.73
Vanilla GPT-2	0.33	0.32
Autoregressive ave.	0.65	0.75

Table 2: Aggregate data for unrestricted and restricted cloze tests. The autoencoding average is shown without BERT-base-it LWYN and BERT-base-multi, while the autoregressive average is shown without Vanilla GPT-2.

4 Case Studies

In this section, we focus on two specific examples of cloze tests from our dataset that serve as case studies to shed further light on our results. Let us consider the following restricted cloze test (G1 in Table 1).

> Nelle frasi seguenti, tratte da un libro di geometria, inserite le parole opportune per mezzo dei menu a discesa.
>
> *Dati due punti distinti A e B esiste una e una sola retta r tale che A e B appartengono [1] r. Invece di ”A appartiene a r” possiamo scrivere ”A giace [2] r” oppure A è un punto [3] r. Due rette complanari hanno o un punto o nessun punto [4] comune. [5] una retta e un punto che non giace [6] medesima, può essere fatto passare uno e un solo piano.*

The replacements are reported in Table 3 and show that this specific cloze test focuses solely on function words.

UmBERTo offers the best performance. UmBERTo's only mistake is at blank 5, where *Tra* is chosen instead of *Per*. We note that this is a typical mistake made by the students who took this cloze

blank	replacement
1	a, su, di, in, per
2	su, a, di, in, per
3	di, a, da, in, per
4	in, a, di, su, per
5	Per, A, Sopra, In, Tra
6	sulla, alla, della, dalla, tra

Table 3: Replacements for example 1.

test. The correct answer, *Per*, ranks second among UmBERTo's top picks, with a probability of approximately $2.9\ 10^{-3}$ as opposed to $3.3\ 10^{-2}$ for *Tra*. The second best models, BERTino, BERT-base, and ELECTRA-base, make an additional mistake at blank 2.

Let us now consider the following unrestricted cloze test (L1 in Table 1).

Ai fini della sicurezza della circolazione e della tutela della vita umana la velocità [1] non può superare i 130 km/h per le autostrade, i 110 km/h per le strade extraurbane principali, i 90 km/h per le strade extraurbane secondarie e per le strade extraurbane locali, e i 50 km/h per le strade nei centri abitati, con la possibilità di [2] il limite fino a un massimo di 70 km/h per le strade urbane le cui caratteristiche costruttive e funzionali lo consentano, [3] installazione degli appositi segnali. Sulle autostrade a tre corsie più corsia di emergenza per ogni senso di marcia, dotate di apparecchiature [4] omologate per il calcolo della velocità media di percorrenza su tratti determinati, gli enti proprietari o concessionari possono elevare il limite massimo di velocítà fino a 150 km/h sulla base delle caratteristiche progettuali ed effettive del tracciato, previa installazione degli appositi segnali, [5] lo consentano l'intensità del traffico, le condizioni atmosferiche prevalenti e i dati di incidentalità dell'ultimo [6]. In caso di precipitazioni atmosferiche di qualsiasi natura, la velocità massima non può superare i 110 km/h per le autostrade e i 90 km/h per le strade extraurbane principali.

The replacements are reported in Table 4 and show that this specific cloze test focuses primarily on content words.

blank	replacement
1	massima
2	elevare
3	previa
4	debitamente
5	purché
6	quinquennio
incorrect	indebitamente, ridurre, finché, secolo, compresa, sebbene, giorno, poiché, esclusa, velocemente, dimezzare, minima

Table 4: Replacements for example 2.

Autoregressive models ace this test. GePpeTto offers the best performance (no incorrect replacements). Recycled GPT-2 is second best, with only one incorrect replacement out of 6: *giorno* is chosen instead of the correct token *quinquennio*. This replacement requires a level of contextual understanding that cannot be realistically expected from a language model at this point in time; our conjecture is that, in this specific instance, GePpeTto's correct replacement is most likely fortuitous (its performance range across all of our tests seems to validate our conjecture). Autoenconding models fare substantially worse, even though ELECTRA and BERT-base are fairly close to the average human performance.

5 Conclusion

While these results are based on as few as eleven cloze tests (and only two unrestricted ones), the key takeaway is that **existing pre-trained Italian language models with no task-specific fine-tuning can successfully tackle (and pass) relatively sophisticated tests** designed for Italian students who have successfully completed their high school education. These results, though preliminary in nature, suggest various research questions, which could be answered based on a larger set of cloze tests. Such questions include whether there exists a pattern to the incorrect replacements made by the models, how the models fare with different parts of speech and with function words as opposed to content words, and how much their performance would improve with task-specific fine-tuning.

References

Amine Abdaoui, Camille Pradel, and Grégoire Sigel. 2020. Load what you need: Smaller versions of mutlilingual bert. In *SustaiNLP / EMNLP*.

Isabella Chiari. 2002. La procedura cloze, la ridondanza e la valutazione della competenza della lingua italiana. *ITALICA*, 79:466–481.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Wietse de Vries and Malvina Nissim. 2020. As good as new. how to successfully recycle english gpt-2 to make models for other languages.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Edoardo Lugarini. 2010. Franco Angeli.

Lorenzo De Mattei, Michele Cafagna, F. Dell'Orletta, M. Nissim, and Marco Guerini. 2020. Geppetto carves italian into a language model. *ArXiv*, abs/2004.14253.

Matteo Muffo and E. Bertino. 2020. Bertino: An italian distilbert model. In *CLiC-it*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

F. W. Radice. 1978. Using the cloze procedure as a teaching technique. *ELT Journal*, XXXII.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

F. Tamburini. 2020. How "bertology" changed the state-of-the-art also for italian nlp. In *CLiC-it*.

Wilson L. Taylor. 1953. 'cloze' procedure: A new tool for measuring readability. *Journalism Quarterly*, 30:415–433.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

KERMIT for Sentiment Analysis in Italian Healthcare Reviews

Leonardo Ranaldi[1]**, Michele Mastromattei**[2]**, Dario Onorati**[2]**,**
Elena Sofia Ruzzetti[2]**, Francesca Fallucchi**[1]**, Fabio Massimo Zanzotto**[2]
1. Dept. of Innovation and Information Engineering Guglielmo Marconi University, Italy
2. Dept. of Enterprise Engineering University of Rome Tor Vergata, Italy
`l.ranaldi@unimarconi.com`, `elenasofia.ruzzetti@alumni.uniroma2.eu`,
{`michele.mastromattei`,`fabio.massimo.zanzotto`}`@uniroma2.it`,
`dario.onorati@uniroma1.it`, `f.fallucchi@unimarconi.it`

Abstract

English. In this paper, we describe our approach to the sentiment classification challenge on Italian reviews in the healthcare domain. Firstly, we followed the work of Bacco et al. (2020) from which we obtained the dataset. Then, we generated our model called KERMIT$_{HC}$ based on *KERMIT* (Zanzotto et al., 2020). Through an extensive comparative analysis of the results obtained, we showed how the use of syntax can improve performance in terms of both accuracy and F1-score compared to previously proposed models. Finally, we explored the interpretative power of *KERMIT*-viz to explain the inferences made by neural networks on examples.

Italiano. *In questo lavoro, presentiamo il nostro approccio al task di sentiment analysis per le recensioni italiane in ambito sanitario. Abbiamo seguito il lavoro di Bacco et al. (2020) da cui abbiamo ottenuto il dataset. Successivamente, abbiamo usato KERMIT$_{HC}$ basato su* KERMIT*(Zanzotto et al., 2020). Da un'ampia analisi comparativa dei risultati ottenuti mostriamo come l'uso della sintassi può migliorare le prestazioni sia in termini di accuratezza che di F1-score rispetto ai modelli proposti in precedenza. Infine, abbiamo esplorato il potere interpretativo di* KERMIT-*viz per spiegare le inferenze fatte dalle reti neurali sugli esempi.*

1 Introduction

People are practically reviewing anything in online sites and understanding the polarization of a comment through automatic sentiment classifier is a tantalizing challenge. In recent years, the number of virtual reviewers has drastically increased and there are many products and services, which can be reviewed. Each person, before buying a product or a service, searches into reviews from people who have already had experienced the product or the service. Review portals are usually linked to the leisure or business activities such as the world of tourism, e-commerce or movies. However, there are topics where these reviews and the associated automatic computed sentiment may induce to select wrong services, which may dramatically affect personal life.

When dealing with health-related services, the effect of positive or negative reviews on hospitals and doctors can have a potential catastrophic impact on the health of who is using this piece of information. QSalute [1] is one of the most important Italian portals of reviews about hospitals, nursing homes and doctors. It is very important for patients to seek the best hospital for their condition based on the past experience of other patients. Reviews in the world of health benefit both patients and hospitals because they are a means to discover problems and solve them (Greaves et al., 2013; Khanbhai et al., 2021).

Automatic sentiment analyzer have then a big responsibility in the context of health-related services. In these sensitive areas, it is important to design AI systems whose decisions are transparent (Doshi-Velez and Kim, 2017), that is, the systems must give the motivation for the choice made so that people can trust. If the users do not trust a

[1] https://www.qsalute.it/

model or a prediction, they will not use it (Ribeiro et al., 2016).

In this article, we investigate a model that can mitigate the responsibility of sentiment analyzers for health-related services. The model we are using exploits syntactic information within neural networks to provide a clear visualisation of the internal decision mechanism of the model that produced the decision. We propose KERMIT$_{HC}$ (*KERMIT* for **H**ealth**C**are) based on *KERMIT* (Zanzotto et al., 2020) to solve the sentiment analysis task introduced by Bacco el al.(2020). We use KERMIT$_{HC}$ on QSalute Italian portal reviews in order to include symbolic knowledge as a part of the architecture and visualize the internal decision-making mechanism of the neural model, using *KERMIT*-viz (Ranaldi et al., 2021).

In the rest of paper, Section 2 gives details about the dataset and methods, while Section 3 and 4 describe the experiments, the results obtained and their discussion. Finally, in Section 5 we present the final conclusions and future goals.

2 Data & Methods

To explore our hunch that syntactic interpretation may help in Healthcare reviews recognition, we leverage: (1) a Healthcare training corpus (Sec. 2.1); (2) a KERMIT$_{HC}$, which is based on syntactic interpretation and it can explain its decisions; and finally, (3) some challenges solved due to KERMIT$_{HC}$ (Sec. 2.2).

2.1 Dataset

In order to investigate reviews in healthcare area, we selected the QSalute portal, one of the most important health websites in Italy. This portal can be defined as the TripAdvisor of hospital facilities, indeed it talks about: *Expertise*, *Assistance*, *Cleaning* and *Services*. In addition to the reviews, there are some associated metadata such as: *user id*, *hospital name*, *review title* and *patient pathology*. To ensure privacy we do not consider sensitive data such as *user id* and *hospital name*.

We used a free available scraper on GitHub [2] to download the dataset. Then, to model this data to a sentiment analysis task, we followed the indications provided by Bacco et al.(2020) - in detail, a review is: (1) negative if the average of its scores is less than or equal to 2, (2) positive if the average of its scores is greater than or equal to 4 (3) neutral otherwise.

The resulting dataset is composed of 47,224 reviews consisting of: 40,641 reviews in the positive class, 3,898 in the neutral class and 2,685 in the negative class.

In this work, we solely consider positive and negative classes, so our final dataset is composed of 43,326 reviews. The dataset is heavily skewed (93,80% positive class - 6,20% negative class) favoring reviews labeled as positive.

2.2 KERMIT 4 Healthcare

KERMIT$_{HC}$ (*KERMIT* for **H**ealth**C**are) architecture is composed of 3 major parts: (1) a *KERMIT* model described in Zanzotto et al. (2020), (2) a Transformers model and (3) a decoder layer that combines the results obtained from the previous two sub-parts. In figure Fig.1 we show a graphical representation of the architecture of KERMIT$_{HC}$, pointing the parts that compose it.

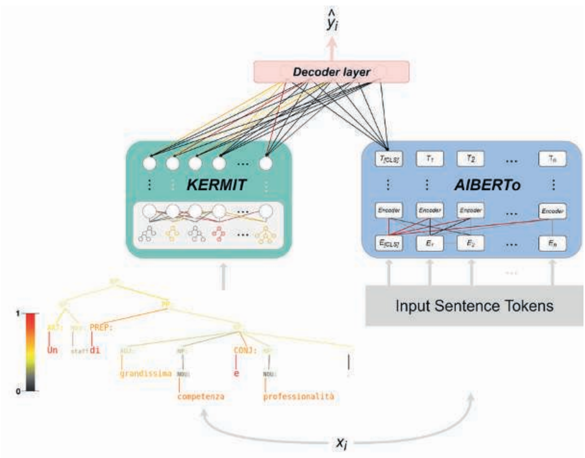


Figure 1: KERMIT$_{HC}$ architecture, forward and interpretation pass.

The architecture of KERMIT$_{HC}$ makes it a particular model, because it combines the syntax offered by KERMIT with the versatility of a Transformer-model. We use KERMIT because it allows the encoding of universal syntactic interpretations in a neural network architecture. KERMIT component is itself composed of two parts: KERMIT encoder, which converts parse tree T into embedding vectors and a multi-layer perceptron that exploits these embedding vectors. The second sub-part of our architecture is composed of a Bidirectional Encoder Representations from Transformers, - as known as BERT - to classify the

[2] The scraper is available at https://github.com/lbacco/Italian-Healthcare-Reviews-4-Sentiment-Analysis

Model	Average Accuracy	Average Macro F1 score	Average Weighed F1 score
UmBERTo	$0.74(\pm0.14)^{\diamond}$	$0.43(\pm0.02)$	$0.75(\pm0.18)^{\circ}$
AlBERTo	$\mathbf{0.82}(\pm\mathbf{0.15})^{\diamond}$	$\mathbf{0.47}(\pm\mathbf{0.05})^{\dagger}$	$\mathbf{0.8}(\pm\mathbf{0.14})^{\circ}$
BERT multilingual	$0.73(\pm0.13)$	$0.46(\pm0.1)^{\dagger}$	$0.73(\pm0.22)$
ELECTRA$_{ita}$	$0.67(\pm0.17)$	$0.4(\pm0.13)$	$0.66(\pm0.2)$

Table 1: Performance of *BERT*, on 25% of the QSalute dataset. Mean and standard deviation results are obtained from 10 runs. For each *Site*, the best performing model was highlighted based on the F1 score values obtained. The symbols $\diamond$, $\circ$ and $\dagger$ indicate a statistically significant difference between two results with a 95% of confidence level with the sign test.

sentiment of the reviews. BERT is a pre-trained language model developed by Devlin et al. (2019) at Google AI Language. In particular, since the task concerns sentences in the Italian language, we have used a special BERT version pretrained on that language called AlBERTo (Polignano et al., 2019).

3 Experiments

We used KERMIT$_{HC}$ architecture to examine if it is possible to answer the research questions showed in *KERMIT* (Zanzotto et al., 2020) also in healthcare domain using the Italian language. Those research questions are: (1) Can the *symbolic knowledge* provided by universal symbolic syntactic interpretations, make a difference and it be used effectively in neural networks? (2) Do *universal symbolic syntactic interpretations* encode different syntactic information than those encoded in *"embeddings of universal sentences"*? (3) Can the *universal symbolic syntactic interpretations* provided by KERMIT$_{HC}$, supply a better and clearer way to explain the decisions of neural networks than those provided by transformers?

To provide a comprehensive answer to these questions, we tested the architecture in a *completely universal* setting where both *KERMIT* and AlBERTo are trained only in the last decision layer.

The rest of the Section describes the experimental set-up, the quantitative experimental results and discusses how we can use the *KERMIT-viz* to explain decisions of neural network inferences over examples.

3.1 Experimental Set-up

This section describes the general experimental set-up of our experiments and the specific configurations adopted.

The parameters used for the *KERMIT* encoder are those proposed in Zanzotto et al., (2020) paper. The constituency parse trees used for *KERMIT* sub-part are obtained using our freely available script on GitHub[3].

We tested several different BERT version pretrained on Italian language in order to get the best model for our task. In particular, we tested the following transformers: (1) UmBERTo (Parisi et al., 2020); (2) AlBERTo (Polignano et al., 2019); (3) BERT multilingual (Devlin et al., 2018) and (4) ELECTRA$_{ita}$: an Italian version of ELECTRA model (Clark et al., 2020) implemented by Schweter (2020) on a work of Chan et al. (2020). All the models were implemented using Huggingface's transformers library (Wolf et al., 2019) and all were used in the uncased setting with the pretrained version. The input text for BERT has been preprocessed and tokenized as specified in respectively work (Parisi et al., 2020; Polignano et al., 2019; Devlin et al., 2018; Schweter, 2020).

Since our experiments are text classification task, the decoder layer of our KERMIT$_{HC}$ architecture is a fully connected layer with the softmax activation function applied to the concatenation of the *KERMIT* sub-part output and the final `[CLS]` token representation of the selected transformer model. Finally, the optimizer used to train the whole architecture is AdamW (Loshchilov and Hutter, 2019) with the learning rate set to $2e^{-5}$. For reproducibility, the source code of our experiments is publicly available on our GitHub repository[4].

[3]The code is available at `https://github.com/LeonardRanaldi/Constituency-Parser-Italian`

[4]The code is available at `https://github.com/ART-Group-it/KERMIT-4-Sentiment-Analysis-on-Italian-Reviews-in-Healthcare`

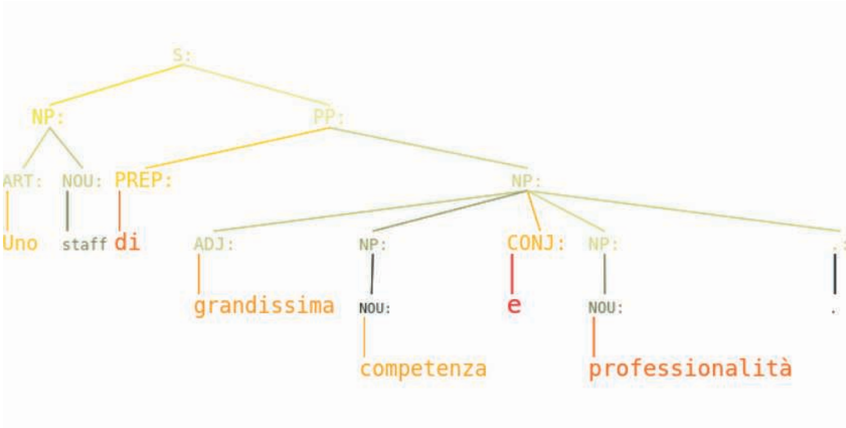
Site	Model	Average Accuracy	Average Macro F1 score	Average Weighed F1 score
Pneumology	**KERMIT$_{HC}$**	**0.71 (± 0.14)**	**0.51 (± 0.08)**	**0.7 (± 0.11)**
	AlBERTo	0.66 (± 0.27)	0.4 (± 0.12)†	0.61 (± 0.26)
Thoracic Surgery	**KERMIT$_{HC}$**	**0.78 (± 0.13)**	**0.51 (± 0.07)**	**0.81 (± 0.08)**
	AlBERTo	0.74 (± 0.28)	0.43 (± 0.13)	0.74 (± 0.26)
Nervous System	**KERMIT$_{HC}$**	**0.87 (± 0.05)**†	**0.6 (± 0.03)**†	**0.89 (± 0.03)**
	AlBERTo	0.94 (± 0.01)†	0.48 (± 0.0)†	0.91 (± 0.01)
Hearth	**KERMIT$_{HC}$**	**0.93 (± 0.03)**†	**0.56 (± 0.03)**†	**0.93 (± 0.02)**
	AlBERTo	0.96 (± 0.01)†	0.49 (± 0.0)†	0.94 (± 0.01)
Vascular Surgery	**KERMIT$_{HC}$**	**0.81 (± 0.16)**	**0.49 (± 0.06)**†	**0.83 (± 0.12)**
	AlBERTo	0.70 (± 0.29)	0.42 (± 0.11)†	0.73 (± 0.23)
Ophthalmology	**KERMIT$_{HC}$**	**0.79 (± 0.08)**	**0.55 (± 0.05)**†	**0.83 (± 0.06)**
	AlBERTo	0.87 (± 0.08)	0.48 (± 0.02)†	0.86 (± 0.04)
Rheumatology	**KERMIT$_{HC}$**	**0.58 (± 0.23)**	**0.43 (± 0.11)**	**0.60 (± 0.20)**
	AlBERTo	0.68 (± 0.20)	0.44 (± 0.10)	0.69 (± 0.19)
Infections	**KERMIT$_{HC}$**	**0.68 (± 0.19)**	**0.51 (± 0.12)**	**0.70 (± 0.17)**
	AlBERTo	0.57 (± 0.23)	0.42 (± 0.13)	0.58 (± 0.21)
Skin	**KERMIT$_{HC}$**	**0.64 (± 0.11)**	**0.50 (± 0.07)**	**0.70 (± 0.10)**
	AlBERTo	0.63 (± 0.26)	0.39 (± 0.11)	0.61 (± 0.24)
Genital	**KERMIT$_{HC}$**	**0.79 (± 0.09)**†	**0.55 (± 0.03)**†	**0.82 (± 0.06)**
	AlBERTo	0.88 (± 0.06)†	0.49 (± 0.02)†	0.87 (± 0.03)
Endoscopy	**KERMIT$_{HC}$**	**0.75 (± 0.09)**	**0.52 (± 0.04)**†	**0.80 (± 0.05)**
	AlBERTo	0.80 (± 0.19)	0.45 (± 0.07)†	0.78 (± 0.17)
Facial	**KERMIT$_{HC}$**	**0.70 (± 0.24)**	**0.42 (± 0.08)**	**0.76 (± 0.18)**
	AlBERTo	0.72 (± 0.26)	0.42 (± 0.10)	0.76 (± 0.22)
Oncology	**KERMIT$_{HC}$**	**0.91 (± 0.06)**	**0.52 (± 0.04)**†	**0.92 (± 0.03)**
	AlBERTo	0.89 (± 0.21)	0.46 (± 0.08)†	0.89 (± 0.17)
Haematology	**KERMIT$_{HC}$**	**0.56 (± 0.30)**	**0.36 (± 0.14)**	**0.57 (± 0.31)**
	AlBERTo	0.41 (± 0.25)	0.30 (± 0.11)	0.46 (± 0.23)
Endocrinology	**KERMIT$_{HC}$**	**0.71 (± 0.20)**	**0.48 (± 0.12)**	**0.71 (± 0.22)**
	AlBERTo	0.73 (± 0.29)	0.41 (± 0.13)	0.69 (± 0.28)
Gynaecology	**KERMIT$_{HC}$**	**0.82 (± 0.08)**	**0.56 (± 0.05)**†	**0.85 (± 0.05)**
	AlBERTo	0.85 (± 0.14)	0.48 (± 0.04)†	0.84 (± 0.09)
Otorhinology	**KERMIT$_{HC}$**	**0.84 (± 0.14)**	**0.50 (± 0.06)**	**0.86 (± 0.09)**
	AlBERTo	0.80 (± 0.18)	0.46 (± 0.05)	0.83 (± 0.13)

Table 2: Performance of KERMIT$_{HC}$ and AlBERTo on QSalute database grouped by *Site*. Mean and standard deviation results are obtained from 10 runs. For each *Site*, the best performing model was highlighted based on the F1 score values obtained. The symbol † indicate a statistically significant difference between two results with a 95% of confidence level with the sign test.

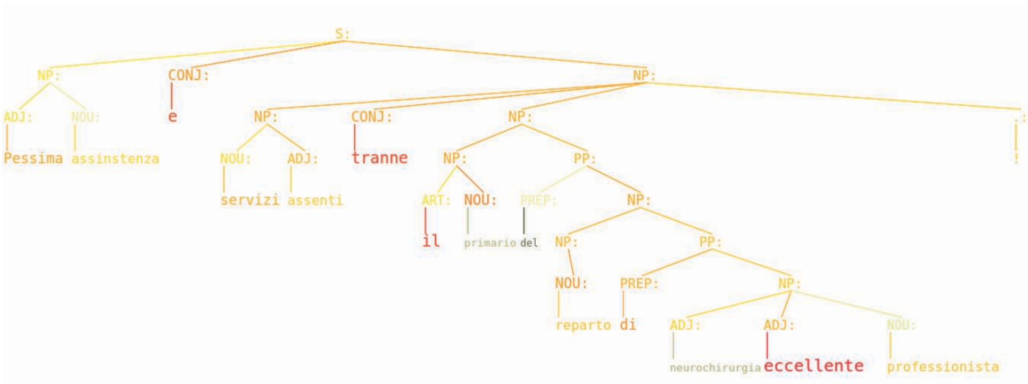
4 Results and Discussion

Syntactic information is useful to significantly increase performances to classify Healthcare reviews (see Table 2). KERMIT$_{HC}$ uses AlBERTo which is the best BERT-italian version model according to our experiments, showed in Table 1. Especially KERMIT$_{HC}$ outperforms the solely AlBERTo sub-part model (ref. to Table 2).

As in the work proposed by Bacco et al.(2020), we chose to divide the dataset by *"Site"* and evaluate the models using accuracy and F1-score metrics. Despite this division, the dataset is still very unbalanced favoring the class 1 (positive reviews). We reports results in terms of the accuracy, Macro F1 and Weighed F1. Observing Table 2, we can see that the performance obtained by KERMIT$_{HC}$ always exceeds the best configuration of *BERT*: AlBERTo. Hence, trained on the Healthcare review dataset (Bacco et al., 2020) (see Section 2.1) KERMIT$_{HC}$ seems to be a good candidate to analyze sentiment of hospital patients.



(a) **S:** `Uno staff di grandissima competenza e professionalità!`



(b) **S:**`Pessima assistenza e servizi assenti tranne il primario di reparto di neurochirurgia eccellente professionista`

Figure 2: The visualizations offered by *KERMIT-viz*. Both examples have the target class positive but in the first one, it is easy to state the positivity. In the second one, who wrote the review, makes disquisitions about the medical staff but at the same time lauds the head of the department.

Using the *KERMIT-viz* visualiser, we analysed how important the contribution of symbolic knowledge provided by *KERMIT* can be. In many cases it makes all the difference. Looking at the Figure 2, these are two sentences with a positive target. The first sentence (shown in Fig. 2a) is clearly positive while the sentence shown in the Fig. 2b could be ambiguous as the patient makes bad remarks about the service but praises the head of the department. We can observe how some words have been colored in red (therefore they have received a greater weight during the classification phase) emphasizing the positive aspects of the sentence and causing it to be labeled as *"positive review"*. In this way the explainability is guaranteed and in very delicate topics - like sentiment in health reviews - we can have more *"trust"* on sentiment analysers.

5 Conclusion

In this article, we investigated a model that can mitigate the responsibility of sentiment analyzers for health-related services. Our model KERMIT$_{HC}$ exploits syntactic information within neural networks to provide a clear visualisation of its internal decision mechanism. KERMIT$_{HC}$ is based on *KERMIT* (Zanzotto et al., 2020) and we worked in a sentiment analysis task introduced by Bacco el al.(2020).

We studied several versions of pre-trained BERT models on the Italian language and found out that AlBERTo is, among them, the best model for this task. However, KERMIT$_{HC}$, which is composed of *KERMIT*+AlBERTo, outperforms better than AlBERTo model alone. Additionally, via *KERMIT-viz*, we visualized the reasons why KERMIT$_{HC}$ classifies the dataset. We observed how KERMIT$_{HC}$ captures relevant syntactic information by catching the keywords in each sentence giving them more weight in the decision phase, mitigating and capturing possible errors of the sentiment analysers. Our future goal is to be able to have full control of the sentiment analysers by injecting human rules (Onorati et al., 2020) in order to mitigate possible errors.

References

Luca Bacco, A. Cimino, L. Paulon, M. Merone, and F. Dell'Orletta. 2020. A machine learning approach for sentiment analysis for italian reviews in healthcare. In *CLiC-it*.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online), December. International Committee on Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning.

Felix Greaves, Daniel Ramirez-Cano, Christopher Millett, Ara Darzi, and Liam Donaldson. 2013. Use of sentiment analysis for capturing patient experience from free-text comments posted online. *Journal of medical Internet research*, 15:e239, 11.

Mustafa Khanbhai, Patrick Anyadi, Joshua Symons, Kelsey Flott, Ara Darzi, and Erik Mayer. 2021. Applying natural language processing and machine learning techniques to patient experience feedback: a systematic review. *BMJ Health & Care Informatics*, 28(1).

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *7th International Conference on Learning Representations, ICLR 2019*.

Dario Onorati, Pierfrancesco Tommasino, Leonardo Ranaldi, Francesca Fallucchi, and Fabio Massimo Zanzotto. 2020. Pat-in-the-loop: Declarative knowledge for controlling neural networks. *Future Internet*, 12(12).

Loreto Parisi, Simone Francia, and Paolo Magnani. 2020. Umberto: An italian language model trained with whole word masking.

Marco Polignano, Pierpaolo Basile, Marco de Gemmis, Giovanni Semeraro, and Valerio Basile. 2019. AlBERTo: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets. In *Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019)*, volume 2481. CEUR.

Leonardo Ranaldi, Francesca Fallucchi, and Fabio Massimo Zanzotto. 2021. KERMITviz: Visualizing Neural Network Activations on Syntactic Trees. In *In the 15th International Conference on Metadata and Semantics Research (MTSR'21)*, volume 1.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?": Explaining the predictions of any classifier.

Stefan Schweter. 2020. Italian bert and electra models, November.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv*, abs/1910.0.

Fabio Massimo Zanzotto, Andrea Santilli, Leonardo Ranaldi, Dario Onorati, Pierfrancesco Tommasino, and Francesca Fallucchi. 2020. KERMIT: Complementing transformer architectures with encoders of explicit syntactic interpretations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 256–267, Online, November. Association for Computational Linguistics.

T-PAS Scraper: an Application for Linguistic Data Extraction and Analysis

Emma Romani[1], **Valerio Gattero**[1], **Elisabetta Ježek**[1]
1. Università degli Studi di Pavia, Pavia, Italy
emma.romani01@universitadipavia.it, valerio.gattero01@universitadipavia.it, jezek@unipv.it

Abstract

In this paper we introduce T-PAS Scraper, a new online application for linguistic data extraction and analysis connected to the T-PAS resource, a corpus-based digital repository of Italian verbal patterns (Ježek et al., 2014). The application is conceived as a supplementation of the main functions of T-PAS and can be used concurrently with the resource, thus extending its accessibility. It consists of 25 different scripts which operate automatically on the database of the resource and can be useful for quantitative and qualitative studies of the linguistic data it contains.

1 Introduction

T-PAS Scraper is a new online application for linguistic data extraction and analysis. It is designed as an extension and supplementation of T-PAS resource (Ježek at al., 2014)[1], a corpus-based digital inventory of Italian verbal patterns, which provide syntactic and semantic information on the verb-argument structures (that is, the patterns)[2].

Initially, the project was not conceived as online application for the linguistic analysis of T-PAS resource data. The first idea was to find a way to speed up the revision of the resource: we needed a fast system that could facilitate the manual correction of annotators' mistakes within the patterns contained in the editor of the resource. As the dimension of the resource is considerable (T-PAS is a repository of 5326 patterns)[3], checking all of them manually while going through the correction and refinement phase would have been time-consuming. Annotators' mistakes are not widespread within the resource as most of the work performed on T-PAS was carried out manually, but isolated errors can occur.

As a solution, we developed a series of T-PAS-specific scripts running on the updated T-PAS resource database (a JSON-structured file containing all the patterns and the related information), which can extract lists of aggregated data, displayed in columns. By skimming the lists, one can easily notice data errors in the extracted data and therefore correct them by moving to the editor of the resource and editing the patterns which were wrongly annotated.

As the number of scripts that we developed was consistent, covering several aspects of the resource, we believed that they would have been useful for users and not only for annotators' revision. We decided to build an online application, which is called T-PAS Scraper, that extends T-PAS accessibility: it can be used by future T-PAS users for quantitative and qualitative studies of its linguistic data[4].

In this paper we describe the application in its components, how it is related to T-PAS resource and some possible uses.

The paper is structured as follows. In Section 2 we briefly describe the T-PAS resource, its main features and the online interface. In Section 3 we introduce T-PAS Scraper and provide a technical

[1] Link to the project: https://tpas.unipv.it/. Both T-PAS resource and T-PAS Scraper are accessible from this link.

[2] T-PAS resource has been developed within Sketch Engine (Kilgarriff et al., 2014); it will be available online by the end of 2021. T-PAS Scraper application has been developed at the University of Pavia during a curricular internship in which the first two authors were involved with the aim of refining, correcting, and improving the resource and its main features before its online publication.

[3] Last update: 25/08/2021.

[4] This is the primary and final purpose of the application, as the refinement, correction, and improvement phase is about to be concluded for the upcoming release of the resource.

explanation on how it was built; we also present the main functions of T-PAS Scraper, and in which sense they are complementary to T-PAS resource, as well as how it can be used from a user perspective. In Section 4 we discuss some future perspective on the project.

2 T-PAS Resource

T-PAS (Ježek et al., 2014) is a corpus-derived resource consisting of an inventory of Typed Predicate-Argument Structures (T-PAS) for Italian verbs. It is a gold standard for Italian verb-argument structures. The resource is being developed at the University of Pavia with the technical support of Lexical Computing Ltd. (CZ) and is intended to be used for linguistic analysis, language teaching, and computational applications. The resource consists of four fundamental components:

1. a repository of corpus-derived predicate argument structures (called *patterns*) with semantic specification of their argument slots, e.g. [Human] drinks [Beverage];
2. an inventory of ca. 200 corpus-derived semantic classes (called *Semantic Types*) organised in a hierarchy (called *System of Semantic Types*), used for the semantic specification of the arguments;
3. a corpus of manually annotated sentences that instantiate the different patterns of the verbs in the inventory. Corpus lines are tagged with their respective pattern numbers and anchored to the verb they feature, which is the lexical unit of analysis[5];
4. an editing system called Skema (Baisa et al., 2020), which allows the registration of patterns and all the syntactic and semantic information associated therewith and facilitates the manual annotation of corpus instances (directly linked to the patterns)[6].

Typed predicate-argument structures are patterns that display the semantic properties of verbs and their arguments: for each meaning of a verb, a specific pattern is provided. As referenced above, the patterns are corpus-derived, i.e., they are acquired through the manual clustering and annotation of corpus instances, following the CPA methodology (i.e., Corpus Pattern Analysis; Hanks, 2013). Currently, T-PAS contains 1165 implemented verbs, 5,326 patterns, and ca. 200,000 annotated corpus instances.

In the resource, each pattern is labelled with a pattern number and connected to a list of corpus instances realising that specific verb meaning. The Skema editor enables the registration of different semantic and lexical information in each pattern: the *verb*, which in T-PAS is generally in its infinitive form - e.g., *bere* (Eng., 'to drink'); the *Semantic Types* (e.g. [Human], [Beverage], always portrayed within square brackets), specifying the semantics of the arguments selected by the verb; argument positions[7], which are filled by the Semantic Types in the patterns; the *sense description*, i.e. a brief definition of the meaning of the verb in that specific pattern; a *lexical set* (optional) for each Semantic Type in the pattern, i.e. a selection of the most representative lexical items instantiating that Semantic Type (e.g. *vino* = 'wine' | *birra* = 'beer' | *aranciata* = 'orange juice' are good candidates for the lexical set of [Beverage]); the *roles* (optional) played by some specific Semantic Types in certain contexts: in particular, the Semantic Type [Human] can acquire the role of Athlete, Doctor, Musician, Host, Guest, Writer, etc., depending on the verb selecting it as an argument; the *features* (optional) associated with the Semantic Types, i.e. certain semantic characteristics required by the pattern syntax (e.g. Plural) or by the specific verb meaning (e.g. Female, Negative, Visible); *prepositions* (for prepositional complements); *particles* (for adverbials); *complementizers* (for clausals); *quantifiers*, and *determiners* (for lexical sets), which can be implemented according to the specific argument position in question.

The System of Semantic Types used to classify the semantics of arguments (Pustejovsky et al., 2004; Ježek, 2019) is a hierarchy of general semantic categories obtained by manual clustering of the lexical items found in the argument positions of corpus-derived valency structures. The System currently contains ca. 200 Semantic Types that are hierarchically organised

[5] The reference corpus for the resource is the web corpus ItWac (reduced), provided by Sketch Engine. It contains around 935 million tokens.

[6] Skema (Baisa et al., 2020) is a corpus pattern editor system implemented to facilitate the management of manual annotation of concordance lines with user-defined labels and the editing of the corresponding patterns in terms of slots, attributes and other features following the lexicographic technique of CPA (Hanks, 2013).

[7] I.e., subject, object, adverbial, clausals, prepositional complement, predicative complement. They can be optional, but yet registered in the pattern if they are relevant to the sense of the verb.

on the basis of the ‘is a’ (subsumption) relation (e.g. [Human] is an [Animate]).

T-PAS online version, which will be publicly available for the users by the end of 2021, will consist of:

1. the repository of predicate-argument structures (patterns);
2. five good corpus examples (GDEX; Kilgarriff et al., 2008) for each of the patterns (previously annotated);
3. the System of Semantic Types;
4. a search engine that allows to search Semantic Types and argument positions (subject, object, etc.) in combination.

T-PAS Scraper aims at completing and integrating T-PAS functionalities: the two interfaces can be used complementarily when visualizing the pattern and searching for specific linguistic phenomena.

3 T-PAS Scraper

3.1 Building T-PAS Scraper

T-PAS Scraper is, in its first release, constituted by two parts: the scripts to retrieve the data from T-PAS database, which were created using Python[8] and PyCharm[9], and a graphical interface that produces a cross-platform executable program.

The program can load a Sketch Engine-compatible database, select a script, and run it. Once clicked on the “Run script” button (see Figure 1), a JSON file is produced with the requested content (i.e., the list of extracted data). Every script is different and prints different complex data in its output, but the way in which data is structured is identical in all of them.

The online user application was created by programming a web application with Angular, a popular front-end framework, PrimeNG, a component library, and Express, a NodeJS server that allows the application to be loaded on Heroku, a hosting company. The procedure followed in order to develop this application consists of two steps: first of all, the scripts were formally defined using a pseudocode, the linguistic data that are the object of the extraction. In the second phase, each script was implemented and printed on a JSON file.

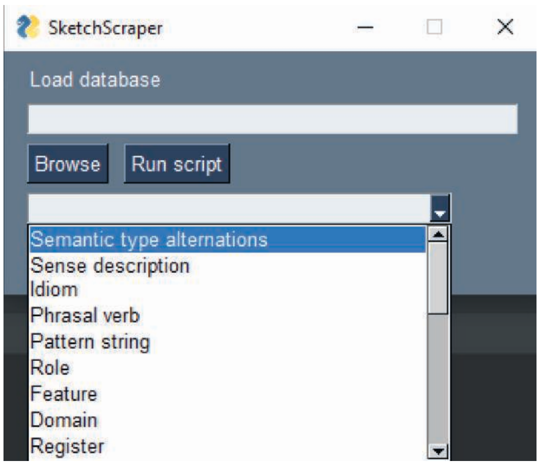


Figure 1. T-PAS Scraper program with the list of scripts that can be run on the JSON database

The online interface (see Figure 2) consists of several parts. First, a menu indicates the top categories of scripts that can been chosen. The top categories are Verbs, Semantic Types, Arguments and Lexical Sets; each of them contains a group of related scripts. For each top category there is a description page and a list of several scripts. Once a script has been selected, the data is shown in table format (generally long lists with different columns), with the possibility of filtering and paging the results in different ways. Data are displayed in alphabetical order, based the first column on the left (containing the verbs). There is also the option to export the current script in Excel format, which can be handy for studying the data externally.

3.2 Using T-PAS Scraper

T-PAS Scraper is useful to access semantic and syntactic information about verbs and their arguments, which cannot be accessed in T-PAS online in aggregate (see Section 2). The basic idea that underlies T-PAS Scraper usage is to have an aggregated overview of the lists (i.e., columns) of data being extracted[10].

As for the verbs, one can visualize the complete list of the 1165 verbs with different information: the number of patterns for each verb (and what is the average number of patterns, as well as which verbs have the highest number of patterns); the frequency and per-million frequency in the corpus (to check whether the frequency is somehow

[8] https://www.python.org/ (last access: 23/09/2021).

[9] https://www.jetbrains.com/pycharm/ (last access: 23/09/2021).

[10] It is possible to refine the research by filtering the columns (e.g., a specific verb and its related information in each script can searched typing the verb in the filter function of the first column of the list, which work as a search box, showing all the items corresponding to the typed query), but this kind of query better fits to T-PAS resource and its search engine (which allows to type a specific verb and open the list of patterns and the examples).

related to the number of patterns of that specific verb). The entire inventory of 5326 patterns contained in T-PAS online, together with the related sense description is provided, both separately and jointly. Verbs in patterns can be registered differently from their base form (i.e., the infinitive) or show for examples reflexive uses, and therefore the entire list of the verb forms can be filtered in order to obtain those forms (e.g., *lavarsi*, Eng. 'to wash yourself').

The complete inventory of the 212 phrasal verb patterns annotated in the resource (e.g., *buttare via*, Eng. 'to throw away') and 388 idiomatic uses (e.g., *bersi il cervello*, 'to go crazy') can also be searched, also in parallel to the patterns in T-PAS online for explanatory examples.

As for Semantic Types, one can search for the most frequent alternations of Semantic Type in argument positions (Ježek et al., 2021; see Figure 2 column 4 for examples) as well as the semantic roles and the features associated to the Semantic Types (see Section 2).

For what concerns the arguments, a list of the argument structures is provided (e.g., subject-object, subject-clausals, subject-prepositional complement) as well as those which are optional and obligatory. Syntactic alternations of arguments (e.g., *finire il pranzo* (object) vs. *finire di mangiare* (clausals) – Eng., 'to finish the meal' vs. 'to finish eating') are also listed.

Finally, complementizers, prepositions, adverbial particles, and obligatory determiners annotated within the patterns in T-PAS, as well as lexical sets, can be extracted through T-PAS Scraper and analysed by the researcher in their distribution.

In Table 1 we provide some quantitative data regarding T-PAS resource, that can be extracted through T-PAS scraper.

script	**n. of items**
verbs	1165
patterns	5326
idiomatic patterns	388
phrasal verb patterns	212
semantic type alternations	3243
semantic types with roles	173
semantic types with features	228
optional arguments	1032
syntactic alternations of arguments	267
patterns with lexical sets	1109

Table 1. Summary of the quantitative data from the scripts

4 Conclusions and Future Perspectives

In this paper we introduced T-PAS Scraper, a new online application for linguistic data extraction and analysis specifically devised to retrieve the data contained in the T-PAS resource and make them available to users for purposes of linguistic analysis, thus extending T-PAS resource accessibility. We described why and how it was

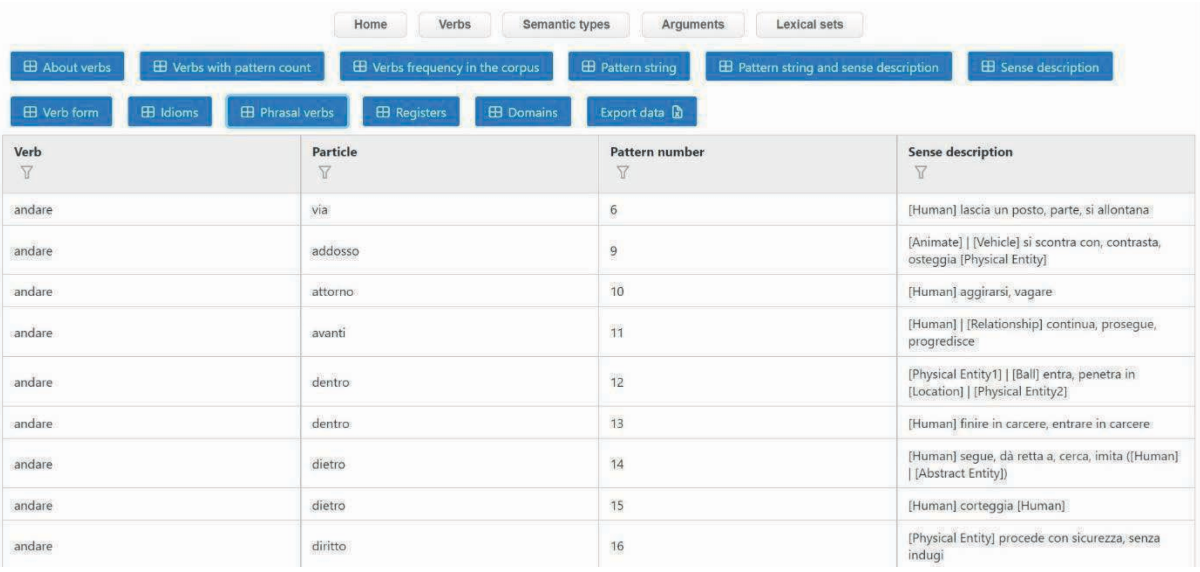


Figure 2. Screenshot of T-PAS Scraper online application with the "phrasal verbs" script as an example (each script has its specific table format as different type of data can be extracted from the resource database). We can see the verb in the first column, the particle in the second, the number of the pattern in the list of available ones for that verb, and the description of the sense of the verb in that pattern.

built and its main functions related to T-PAS resource. We also suggested some possible uses in terms of qualitative and quantitative analysis, from a user perspective.

As a new-born project, T-PAS Scraper application is just at its initial stage and further work can be done. In particular, new scripts can be added to enrich the existing ones.

Currently, the data displayed on the application are in a static form: the updated database needs to be manually re-uploaded in case of some changes in T-PAS editor. The final goal would be to load the data from T-PAS database within Sketch Engine in real time, run the scripts and show the results: in a first phase the data will be displayed in this way. Loading data directly from Sketch Engine also requires coordination and the creation of a dedicated API with external authentication, which does not currently exist. A dynamic infrastructure has countless advantages: it allows to view data in real time, it is scalable and functional, and can also communicate with other systems.

T-PAS Scraper may eventually be extended to resources other than T-PAS whose structure is compatible with the database configuration of T-PAS and Sketch Engine.

References

Baisa, V., Tiberius, C., Ježek, E., Colman, L., Marini, C. & Romani, E. (2020). Skema: A New Tool for Corpus-driven Lexicography. In *Proceedings of the 19th EURALEX International Congress*.

Hanks, P. (2013). *Lexical Analysis: Norms and Exploitations*. Cambridge: The MIT Press.

Ježek, E. (2019). Sweetening Ontologies Cont'd: Aligning bottom-up with top-down ontologies. In A. Barton, S. Seppälä & D. Porello (eds.) *Proceedings of the Joint Ontology Workshops 2019*. Graz, Austria.

Ježek, E., Magnini, B., Feltracco, A., Bianchini, A. & Popescu, O. (2014). T-PAS: A resource of corpus-derived Types Predicate-Argument Structures for linguistic analysis and semantic processing. In *Proceedings of LREC*. pp. 890–895.

Ježek, E., Marini, C., Romani, E. (2021). Encoding semantic phenomena in verb-argument combinations. In Kosem, I., Cukr, M., Jakubíček, M., Kallas, J., Krek, S. & Tiberius, C. (eds.), *Electronic lexicography in the 21st century. Proceedings of the eLex 2021 conference*. 5–7 July 2021, virtual. Brno: Lexical Computing CZ, s.r.o.

Kilgarriff, A., Baisa, V., Bušta, J., Jakubíček, M., Kovář, V., Michelfeit, J., Rychlý, P. & Suchomel, V. (2014). The Sketch Engine: ten years on. Lexicography, 1(1), 7-36.

Kilgarriff, A., Husák, M., McAdam, K., Rundell, M. & Rychlý, P. (2008, July). GDEX: Automatically finding good dictionary examples in a corpus. In *Proceedings of the XIII EURALEX international congress*. Barcelona, Spain: Documenta Universitaria. pp. 425-432.

Pustejovsky, J., Hanks, P. & Rumshisky, A. (2004). Automated Induction of Sense in Context. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*. Geneva, Switzerland.