Proceedings
of the
Fifth Italian Conference
on
Computational Linguistics
CLiC-it 2018

10-12 December 2018, Torino

Editors:

Elena Cabrio
Alessandro Mazzei
Fabio Tamburini

aA

Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it 2018

10-12 December 2018, Torino

Elena Cabrio, Alessandro Mazzei and Fabio Tamburini (dir.)

DOI: 10.4000/books.aaccademia.2802
Publisher: Accademia University Press
Place of publication: Torino
Year of publication: 2018
Published on OpenEdition Books: 8 April 2019
Series: Collana dell'Associazione Italiana di Linguistica Computazionale
Electronic EAN: 9788831978682

http://books.openedition.org

Printed version
Number of pages: 382

Electronic reference
CABRIO, Elena (ed.) ; MAZZEI, Alessandro (ed.) ; and TAMBURINI, Fabio (ed.). *Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it 2018: 10-12 December 2018, Torino.* New edition [online]. Torino: Accademia University Press, 2018 (generated 21 septembre 2021). Available on the Internet: <http://books.openedition.org/aaccademia/2802>. ISBN: 9788831978682. DOI: https://doi.org/10.4000/books.aaccademia.2802.

Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it 2018

10-12 December 2018, Torino

Editors:

Elena Cabrio
Alessandro Mazzei
Fabio Tamburini

aA accademia university press

aA

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

isbn 978-88-31978-41-5
www.aAccademia.it/CLIC_2018

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

Table of Contents

Associazione Italiana di Linguistica Computazionale

Preface

On behalf of the Program Committee, a very warm welcome to the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018). This edition of the conference is held in Torino. The conference is locally organised by the University of Torino and hosted into its prestigious main lecture hall "*Cavallerizza Reale*". The CLiC-it conference series is an initiative of the Italian Association for Computational Linguistics (AILC) which, after five years of activity, has clearly established itself as the premier national forum for research and development in the fields of Computational Linguistics and Natural Language Processing, where leading researchers and practitioners from academia and industry meet to share their research results, experiences, and challenges.

This year CLiC-it received 70 submissions against 64 submissions in 2015, 69 in 2016 and 72 in 2017. The Programme Committee worked very hard to ensure that every paper received at least two careful and fair reviews. This process finally led to the acceptance of 18 papers for oral presentation and 45 papers for poster presentation, with a global acceptance rate of 90% motivated by the inclusive spirit of the conference. The conference is also receiving considerable attention from the international community, with 16 (23%) submissions showing at least one author affiliated to a foreign institution. Regardless of the format of presentation, all accepted papers are allocated 5 pages plus 2 pages for references in the proceedings, available as open access publication. In line with previous editions, the conference is organised around thematic areas managed by one or two area chairs per area.

In addition to the technical programme, this year we are honoured to have as invited speakers internationally recognised researchers as Johan Bos (University of Groningen) and Iryna Gurevych (Technische Universität Darmstadt). We are very grateful to Johan and Iryna for agreeing to share with the Italian community their knowledge and expertise on key topics in Computational Linguistics.

Traditionally, around one half of the participants at CLiC-it are young postdocs, PhD students, and even undergraduate students. As in the previous edition of the conference, we organised a special track called "Research Communications", encouraging authors of articles published in 2018 at outstanding international conferences in our field to submit short abstracts of their work. Research communications are not published in the proceedings, but are orally presented within a dedicated session at the conference, in order to enforce dissemination of excellence in research.

Moreover, during the conference we award the prize for the best Master Thesis (*Laurea Magistrale*) in Computational Linguistics, submitted at an Italian University between August 1st 2017 and July 31st 2018. This special prize is also endorsed by AILC. We have received 6 candidate theses, which have been evaluated by a special jury. The prize will be awarded at the conference, by a member of the jury.

As last year, we propose a tutorial at the beginning of the conference (Paolo Rosso – Profiling Information in Social Media). We highlight the importance that this kind of opportunities have for young researchers in particular, and we are proud of having made the tutorial attendance free for all registered students.

Even if CLiC-it is a medium size conference, organizing this annual meeting requires major effort from many people. This conference would not have been possible without the dedication, devotion and hard work of the members of the Local Organising Committee, who volunteered their time and energies to contribute to the success of the event. We are also extremely grateful to our Programme Committee members for producing a lot of detailed and insightful reviews, as well as to the Area Chairs who assisted the Programme Chairs in their duties. All these people are named in the following pages. We also want to acknowledge the support from endorsing organisations and institutions and from all of our sponsors, who generously provided funds and services that are crucial for the realisation of this event. Special thanks are also due to the University of Torino for its support in the organisation of the event and for hosting the conference at the main lecture hall "*Cavallerizza Reale*".

Please join us at CLiC-it 2018 to interact with experts from academia and industry on topics related to Computational Linguistics and Natural Language Processing and to experience and share new research findings, best practices, state-of-the-art systems and applications. We hope that this year's conference will be intellectually stimulating, and that you will take home many new ideas and methods that will help extend your own research.

Elena Cabrio, Alessandro Mazzei and Fabio Tamburini
CLiC-it 2018 General Chairs

Keynote Talks

Computational Semantics in Neural Times

Johan Bos
University of Groningen, Netherlands
johan.bos@rug.nl

Abstract

Semantic parsing is more popular than ever. One reason is that we have a rising number of semantically annotated corpora. Another reason is that there is new AI technology to be explored. In this talk I will present a new corpus of open-domain texts annotated with formal meaning representations. Using a parallel corpus, the resource is developed not only for English, but also for Dutch, German and Italian. The meaning representations comprise logical operators to assign scope, comparison operators, and non-logical symbols. The non-logical symbols are completely grounded in WordNet concepts and VerbNet-style roles. I will contrast two methods for semantic parsing on this corpus: a traditional technique using a categorial grammar and lambda-calculus, and an ultra-modern way using a (surprise, surprise) neural network. Guess which one performs better!

Short Bio

Johan Bos is Professor of Computational Semantics at the University of Groningen. He received his doctorate from the Computational Linguistics Department at the University of the Saarland in 2001. Since then, he held post-doc positions at the University of Edinburgh, working on spoken dialogue systems, and the La Sapienza University of Rome, conducting research on automated question answering. In 2010 he moved to his current position in Groningen, leading the computational semantics group. Bos is the developer of Boxer, a state-of-the-art wide-coverage semantic parser for English, initiator of the Groningen Meaning Bank, a large semantically-annotated corpus of texts, and inventor of Wordrobe, a game with a purpose for semantic annotation. Bos received a €1.5-million Vici grant from NWO in 2015 to investigate the role of meaning in human and machine translation.

Disentangling the Thoughts: Latest News in Computational Argumentation

Iryna Gurevych
Technische Universität Darmstadt, Germany
gurevych@ukp.informatik.tu-darmstadt.de

Abstract

In this talk, I will present a bunch of papers on argument mining (co-)authored by the UKP Lab in Darmstadt. The papers have appeared in NAACL, TACL and related venues in 2018. In the first part, I will talk about large-scale argument search, classification and reasoning. In the second part, the focus will be on mitigating high annotation costs for argument annotation. Specifically, we tackle small-data scenarios for novel argument tasks, less-resourced languages or web-scale argument analysis tasks such as detecting fallacies. The talk presents the results of ongoing projects in Computational Argumentation at the Technische Universität Darmstadt [1]: Argumentation Analysis for the Web (ArguAna) [2], Decision Support by Means of Automatically Extracting Natural Language Arguments from Big Data (ArgumenText) [3].

[1] https://www.ukp.tu-darmstadt.de/research/research-areas/argumentation-mining/
[2] https://www.ukp.tu-darmstadt.de/research/current-projects/arguana/
[3] https://www.argumentext.de/

Short Bio

Iryna Gurevych is professor of computer science at TU Darmstadt, where she leads the UKP Lab and the DFG-funded Research Training Group "Adaptive Preparation of Information from Heterogeneous Sources" (AIPHES). She has a broad range of research interests in natural language processing, with a focus on computational argumentation, computational lexical semantics, semantic information management, and discourse and dialogue processing. She has co-founded and co-organized the workshop series "Collaboratively Constructed Semantic Resources and their Applications to NLP", "Argument Mining" and several research events on innovative applications of NLP to education, social sciences and humanities. More information can be found: https://www.ukp.tu-darmstadt.de/ .

Contributed Papers

A distributional study of negated adjectives and antonyms

Laura Aina[*]
Universitat Pompeu Fabra
Barcelona, Spain
laura.aina@upf.edu

Raffaella Bernardi
University of Trento
Trento, Italy
bernardi@disi.unitn.it

Raquel Fernández
University of Amsterdam
Amsterdam, The Netherlands
raquel.fernandez@uva.nl

Abstract

English. In this paper, we investigate the relation between negated adjectives and antonyms in English using Distributional Semantics methods. Results show that, on the basis of contexts of use, a negated adjective (e.g., *not cold*) is typically more similar to the adjective itself (*cold*) than to its antonym (*hot*); such effect is less strong for antonyms derived by affixation (e.g., *happy - unhappy*).

Italiano. *In questo lavoro, analizziamo la relazione fra aggettivi negati e antonimi in inglese utilizzando metodi di Semantica Distribuzionale. I risultati mostrano che, sulla base dei contesti di uso, la negazione di un aggettivo (ad es. "not cold"; it.: "non freddo") è tipicamente più simile all'aggettivo stesso ("cold"; it.: "freddo") che al suo antonimo ("hot"; it.: "caldo"). Tale effetto è meno accentuato per antonimi derivati tramite affissi (ad es. "happy"-"unhappy"; it.: "felice"-"infelice").*

1 Introduction

Negation has long represented a challenge for theoretical and computational linguists (see Horn (1989) and Morante and Sporleder (2012) for overviews): in spite of the relative simplicity of logical negation ($\neg p$ is true $\leftrightarrow$ p is false), complexity arises when negation interacts with morphosyntax, semantics and pragmatics.

In this work, we focus on the negation of adjectives in English, expressed by the particle *not* modifying an adjective, as in *not cold*. A naïve account of these expressions would be to equate them to antonyms, and hence take them to convey the opposite of the adjective (e.g., *not cold* = *hot*). In fact, this simplifying assumption is sometimes made in computational approaches which model negation as a mapping from an adjective to its antonym (e.g., The Pham et al., (2015), Rimell et al., (2017)). However, a range of studies support what is known as *mitigation hypothesis* (Jespersen, 1965; Horn, 1972; Giora, 2006), according to which a negated adjective conveys an intermediate meaning between the adjective and its antonym (e.g., *not large* $\approx$ *medium-sized*). The meaning of the adjective is mitigated by negation, while some emphasis on it still persists in memory (Giora et al., 2005). This view is compatible with pragmatic theories predicting that the use of a more complex expression (*not large*) when a simpler one is available (*small*) triggers the implicature that a different meaning is intended (e.g., *medium-sized*) (Grice, 1975; Horn, 1984). Computational models predicting similar mitigating effects are those by Hermann et al., (2013) and Socher et al., (2012; 2013).

In this work, we investigate negated adjectives from the perspective of Distributional Semantics (Lenci, 2008; Turney and Pantel, 2010). We study antonymic adjectives and their negations in terms of their distribution across contexts of use: to this end, we employ an existing dataset of antonyms, whose annotation we further extend, and the distributional representations of these and their negated version, as derived with a standard distributional model. This allows us to conduct a data-driven study of negation and antonymy that covers a large set of instances. We compare pairs of antonyms with distinct lexical roots and those derived by affixation, i.e., **lexical and morphological antonyms** (Joshi, 2012) (e.g., *small - large* and *happy - unhappy* respectively). More-

[*] Part of the work presented in this paper was carried out while the first author was at the University of Amsterdam.

over, we investigate the distinction between lexical antonyms that are **contrary or contradictory**, that is, those that have or do not have an available intermediate value (Fraenkel and Schul, 2008): e.g., something *not cold* is not necessarily *hot* - it could be *lukewarm* - but something *not present* is *absent*. As for negations of morphological antonyms, we compare instances of **simple and double negation**, where the latter occurs if the antonym that is negated is an affixal negation (e.g., *not unhappy*).

Our analyses show that, when considering distributional information, negated adjectives are more similar to the adjective itself than to the antonym (e.g., *not cold* is closer to *cold* than to *hot*), regardless of the type of antonym or of negation. However, we find that morphological antonymy is closer to negation than lexical one is.

2 Motivation and data

We are interested in how negation acts with respect to pairs of adjectives connected by the lexical relation of antonymy (Murphy, 2003), i.e., that are associated with opposite properties within the same domain (e.g., *hot* - *cold*). In particular, we want to compare the negation of one of the antonymic adjectives with itself and its antonym respectively (e.g., *not cold* vs. *cold* and vs. *hot*). Our data of interest are then triples obtained starting from an antonymic pair and negating one of the two items (for each pair we obtain two triples). For example:

(1) $\langle$ *hot*, *cold*, *not* $\{$*hot*$|$*cold*$\}\rangle$

(2) $\langle$*happy*, *unhappy*, *not* $\{$*happy*$|$*unhappy*$\}\rangle$

As data, we make use of a subset of the Lexical Negation Dictionary by Van Son et al. (2016). This consists of antonym pairs in WordNet (Fellbaum, 1998) annotated for different types of lexical negation (Joshi, 2012). We consider adjective pairs that are either *lexical* antonyms, i.e., with distinct lexical roots (e.g., *cold* - *hot*), or *morphological* antonyms, i.e., derived by affixal negation (e.g., *happy* - *unhappy*).[1] In our analyses, we compare different subsets of the data: we explicate and motivate the distinctions in the following.

Lexical vs. morphological antonyms These two groups are usually taken to express the same lexical relation - i.e., opposition - and to be different only on morphological terms. However, such difference might affect their relation with negated adjectives: indeed, affixal negations have a morphological structure that resembles negated adjectives (e.g., *un-happy* vs. *not happy*). For this reason, we keep triples derived from lexical and morphological antonyms distinct, and compare them in our analyses: in particular, we are interested in testing whether in a distributional space negation tends to be more similar to morphological antonymy than to lexical one. Besides this comparison, we apply other distinctions to the triples obtained with lexical and morphological antonyms respectively, in order to investigate further effects.

	adj.	not_adj.	# triples
Lexical antonyms	254715	1144	**198**
– contrary	336923	1057	68
– contradictory	298378	1031	28
Morphological antonyms	83232	1821	**185**
– simple negations	84744	2002	157
– double negations	122525	871	28

Table 1: Average frequency of adjectives and negated adjectives per class, and total number of triples $\langle a_1, a_2, not\ \{a_1|a_2\}\rangle$ considered.

Contrary vs. contradictory Lexical antonyms have been classified as either *contradictory* or *contrary* (Clark, 1974), depending on whether the negation of one entails the truth of the other, without the availability of a mid-value. Fraenkel and Shul (2008) provided psycholinguistic results showing that if an adjective is part of a contradictory pair, its negation is interpreted as closer to the antonym than if it is part of a contrary pair (e.g., *not dead* is interpreted as being closer to *alive* than *not small* to *large*). We aim to investigate this result in a distributional space, where we are able to quantify similarities between lexical items.

Since no data annotated with respect to this distinction is available, the three authors independently annotated the antonym pairs in the dataset as either contrary, contradictory or unclear, following the definition used by Fraenkel and Shul (2008).[2] Not surprisingly, the inter-annotator agreement is only moderate (Fleiss' $k = 0.37$): already Fraenkel and Shul (2008) noted that even for what they considered contradictory pairs it is possible to conceive a mid-value interpretation (e.g., *not dead* $\approx$ *half-dead*; Paradis and Willners (2006)). This suggests that the contrary

[1] In the dataset, the former are coded as *regular antonyms* and the latter as *direct affixal negations*.

[2] Annotation guidelines at `https://lauraina.github.io/data/notadj.pdf`

vs. contradictory distinction involves a continuum rather than a dichotomy. We leave this aspect to be further clarified by future research and, for the purpose of our analysis, only consider pairs classified with full agreement.

Simple vs. double negation In the case of morphological antonyms, one of the two adjectives is an affixal negation, and hence already contains a negating prefix (such as *un-* in *unhappy*): adding *not* thus gives rise to a double negation (e.g., *not unhappy*). These expressions have been widely studied in the literature due to their difference with double negation in logic (e.g., Bolinger (1972), Krifka (2007) and recently Tessler and Franke (2018)). While in logic two negations cancel each other out ($\neg\neg p \equiv p$), in natural language double negations are typically employed to weaken the meaning of the adjective that is negated twice (e.g., *not unhappy* $\neq$ *happy*) . Our goal is to test whether evidence for this effect is found in a distributional space: in particular, if two negations were to cancel each other out then the negation of an affixal negation (e.g., *not unhappy*) should be particularly close to the antonym (e.g., *happy*). We then test whether simple (e.g., *not happy*) and double (e.g., *not unhappy*) negations exhibit similar trends in relation to an antonym pair (*happy* vs. *unhappy*).

3 Analyses

3.1 Methods

Previous studies about negation of adjectives described its effect as a meaning shift from the adjective towards the antonym, that can be measured in terms of semantic similarity (Fraenkel and Schul, 2008). Distributional Semantics offers us a data-driven method of quantifying this: we can represent expressions as vectors summarizing their large-scale patterns of usage and then interpret their proximity relations in terms of similarity.

To this aim, we build a distributional semantic model with standard techniques, but whose vocabulary includes, besides word units, also negated adjectives. In practice, each occurrence of a negated adjective (adjacent occurrence of *not* and an adjective without intervening words; e.g., we exclude cases like *not very cold*) is treated as a single and independent token (e.g., *not cold* $\rightsquigarrow$ *not_cold*). With this pre-processing, we train a word2vec CBOW model (Mikolov et al., 2013)[3] on the concatenation of UkWaC and Wackypedia-En corpora (2.7B tokens; Baroni et al., (2009)), setting parameters as in the best performing model by Baroni et al. (2014).[4] We do not carry out any hyperparameters search, nor we employ any ad hoc techniques aimed at, for example, amplifying the distances between antonyms in the semantic space (such as that of Nguyen et al. (2016) or The Pham et al. (2015)). Indeed, we are interested in investigating characteristics of antonyms and negated adjectives in a standard distributional model, that is not fine-tuned to a particular task and where no assumptions about the structure of its space are incorporated. However, we assess the quality of the induced model through a similarity relatedness task, where we find that it achieves satisfying performances.[5]

For our analyses, we consider triples as those described in Section 2. Given a triple $\langle a_i, a_j, not\ a_i \rangle$ (e.g., *cold*, *hot*, *not cold*), we define the following score:

(3) $\quad \textit{Shift} := \textit{Sim}(\text{not } a_i, a_j) - \textit{Sim}(\text{not } a_i, a_i)$

where $i \neq j$, and $\textit{Sim}(\text{not } a_i, a_j)$ and $\textit{Sim}(\text{not } a_i, a_i)$ are the cosine similarities of the negated adjective with the antonym and the adjective, respectively. This measures how much closer a negated adjective is to the antonym than to the adjective (i.e., how much closer *not cold* is to *hot* than to *cold*), and hence how much negation shifts the meaning of an adjective towards that of the antonym. Due to the well-known tendency of antonyms to be close in a distributional space (Mohammad et al., 2013), the absolute value of *Shift* is not expected to be high (a vector close to one is likely close to the other too). However, we can test whether a higher proximity is registered towards one of the two adjectives.

From the data introduced in Section 2, we only consider triples where each of the three elements occurs at least 100 times in the training corpus of the distributional model. Table 1 shows the number of triples considered for each class and the average frequency of adjectives and negated adjectives.[6] The number of contradictory triples is

[3] Gensim implementation.

[4] Vectors size: 400; window size: 5; minimum frequency: 20; sample: 0.005; negative samples: 1.

[5] Spearman's ρ of 0.75 on the MEN dataset (Bruni et al., 2014); see results by Baroni et al. (2009) for a comparison.

[6] Negated adjectives are overall less frequent than their non-negated counterparts, as shown in Table 1.

small due to the choice of keeping only antonyms for which we had full agreement in the annotation; double negations triples are few due to the limited frequency of these expressions in the corpus.[7]

3.2 Results and discussion

Table 2 shows the scores across the different categories mentioned in Section 2. Example triples for each category are given in Table 3, together with the nearest adjectives of each element in the triple.

Lexical vs. morphological antonyms The average *Shift* scores of both classes are negative, showing that a negated adjective is typically closer to the adjective than to the antonym. Indeed, as shown in Table 3, the nearest neighbor of a negated adjective is often the related adjective. On one hand, this could be seen as supporting the idea that negated adjectives express an intermediate meaning between that of the adjective and the antonym (e.g., *not small* is close to *normal-sized*). More in general, it shows that negated adjectives have a profile of use that is more similar to that of the adjective than to the antonym.

The two classes of antonyms differ significantly in the extent of this effect: negated adjectives are closer to a morphological antonym than a lexical one (e.g., *not perfect* vs. *imperfect*, *not wide* vs. *narrow*). Such similarity in distribution can be explained by the similarity in structure, and hence possibly in meaning, of negated adjectives and affixal negations. Yet, in spite of the higher similarity in use, affixal negation still does not seem equivalent to negation by *not*, due to the negative average *Shift* value.

Contrary vs. contradictory antonyms In contrast to the results from the linguistic literature (see Section 2), the behavior of contrary and contradictory antonym pairs is not significantly different in our analysis. When we look into a distributional space, even for contradictory antonyms, the negated adjectives tend to be more similar to the adjective itself than to the antonym.

This result points at the fact that distributional similarity is capturing a different type of similarity from that considered in the experiments of Fraenkel and Shul (2008). We cannot thus directly interpret our results as just a product of the mitigating aspect of negation. Distributional information may discriminate between the negation of an adjective and the antonym, even when the two seem intuitively equivalent (e.g., *not dead* is closer to *dead* than to *alive*): indeed, the use of one or the other may serve different functions (e.g., contradicting an expectation, politeness, etc.), leading them to appear in different contexts. Moreover, we find that, since continuous representations are able to capture nuanced differences, the alleged dichotomy between contrary and contradictory antonyms may become a continuum in distributional space: for example, one of the closest adjectives to *not dead* is *half-dead*. This further underscores the difficulty in distinguishing between contrary and contradictory antonyms which we had already encountered in the annotation.

Simple vs. double negations There is not a significant difference between negated adjectives that are instances of simple and double negations: crucially, it is not the case that double negations are very close to the antonym as a result of the two negations canceling each other out (e.g., *not unhappy* is closer to *unhappy* than to *happy*).

As before, the result cannot be interpreted only in terms of mitigation (though, e.g., *not unhappy* is close to *unimpressed*, hence a mid-value between *happy* and *unhappy*). In general, it suggests that the contexts of use of double negations are more similar to the ones of the adjective that is negated than to those of its antonym. Indeed, double negations typically appear in contexts where the use of the "logically" equivalent alternative (i.e., the antonym) is to be avoided for pragmatic reasons, as possibly too strong or direct (e.g., *not unproblematic* vs. *problematic*; Horn, (1984)).

4 Conclusion

We have investigated negated adjectives using the tools of Distributional Semantics, which allows us to quantify the similarities between expressions on the basis of how they are used. Our analyses show that, when considering contexts of occurrence, negating an adjective does not make it closer to the antonym than to the adjective itself. This can be seen as a result of the various functions of negation (e.g., mitigation, contradiction to an expectation, politeness) that may lead to different patterns of use for negated adjectives and antonyms. Further research may shed light on which type of contexts actually discriminate them, for example through a corpus study, and which other properties negated adjectives have in a distri-

[7] Full list of triples at `https://lauraina.github.io/data/notadj.pdf`

Lexical antonyms	$-.19$ ($\sigma = .16$)	Morphological antonyms	$-.04$ ($\sigma = .16$)	***
Contrary antonyms	$-.18$ ($\sigma = .15$)	Contradictory antonyms	$-.19$ ($\sigma = .16$)	
Simple negations	$-.03$ ($\sigma = .17$)	Double negations	$-.06$ ($\sigma = .11$)	

Table 2: Average *Shift* scores, with standard deviation, for each category. ***: significant difference between categories in the row ($p < 0.001$, Welch's t-test).

Contrary antonyms	**small**: *large, tiny, smallish, sizeable, largish*	**large**: *small, sizeable, huge, vast, smallish*	**not small**: *small, smallish, normal-sized, largish, middle-sized*
Contradictory antonyms	**dead**: *drowned, lifeless, half-dead, wounded, alive*	**alive**: *dead, awake, unharmed, beloved, tortured*	**not dead**: *dead, half-dead, alive, comatose, lifeless*
Simple negations	**similar**: *analogous, identical, comparable, dissimilar, same*	**dissimilar**: *similar, different, distinct, unrelated, identical*	**not similar**: *similar, dissimilar, identical, distinguishable, analogous*
Double negations	**happy**: *glad, pleased, contented, nice, kind*	**unhappy**: *disappointed, dissatisfied, unsatisfied, resentful, anxious*	**not unhappy**: *unhappy, adamant, disappointed, dismayed, unimpressed*

Table 3: Nearest adjectives is semantic space for the three elements in some sample triples.

butional space, such as their interaction with scalar dimensions (e.g., *not hot* vs. *freezing, cold, lukewarm, hot* etc.; Wilkinson and Tim (2016)). Finally, while for the purpose of this study we opted for a standard word2vec model, one could test for the same effects with differently obtained distributional vectors.

Despite its current limitations in covering truth-related aspects of meaning, Distributional Semantics was shown by Kruszewski et al. (2017) to be apt to model at least some aspects of negation, especially if graded in nature, such as alternativehood. Our study provides supporting evidence for this line of research and in addition points at the utility of using Distributional Semantics to uncover nuanced differences in use between a negation and other expressions, even when logically equivalent. Moreover, we regard our results to be of general interest for the NLP community, since effects of negation like the ones we studied and how they are represented in a distributional space can be critical for tasks like sentiment analysis (e.g., what does it imply that a costumer is *not happy* or *not unhappy* with a product?; Wiegand et al, (2010)).

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 715154), and by the Catalan government (SGR 2017 1575). This paper reflects the authors' view only, and the EU is not responsible for any use that may be made of the information it contains.

References

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

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 238–247.

Dwight Bolinger. 1972. *Degree words*. Walter de Gruyter.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 49(2014):1–47.

Herbert H. Clark. 1974. Semantics and comprehension. In Thomas A. Sebeok, editor, *Current trends in linguistics: Linguistics and adjacent arts and sciences*, volume 12, pages 1291–1428. Mouton.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT press.

Tamar Fraenkel and Yaacov Schul. 2008. The meaning of negated adjectives. *Intercultural Pragmatics*, 5(4):517–540.

Rachel Giora, Noga Balaban, Ofer Fein, and Inbar Alkabets. 2005. Negation as positivity in disguise. In Albert N. Katz and Herbert L. Colston, editors, *Figurative language comprehension: Social and cultural influences*, pages 233–258. Lawrence Erlbaum Associates.

Rachel Giora. 2006. Anything negatives can do affirmatives can do just as well, except for some metaphors. *Journal of Pragmatics*, 38(7):981–1014.

H. Paul Grice. 1975. Logic and conversation. *Syntax and Semantics*, pages 41–58.

Karl Moritz Hermann, Edward Grefenstette, and Phil Blunsom. 2013. "Not not bad" is not "bad": A distributional account of negation. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, pages 74–82.

Laurence R. Horn. 1972. *On the Semantic Properties of Logical Operators in English*. University of California, Los Angeles.

Laurence R. Horn. 1984. Toward a new taxonomy for pragmatic inference: Q-based and R-based implicature. *Meaning, form, and use in context: Linguistic applications*, pages 11–42.

Laurence R. Horn. 1989. *A natural history of negation*. University of Chicago Press.

Otto Jespersen. 1965. *The philosophy of grammar*. University of Chicago Press.

Shrikant Joshi. 2012. Affixal negation: direct, indirect and their subtypes. *Syntaxe et semantique*, 13(1):49–63.

Manfred Krifka. 2007. Negated antonyms: Creating and filling the gap. In Uli Sauerland and Penka Stateva, editors, *Presupposition and implicature in compositional semantics*, pages 163–177. Palgrave McMillan.

Germán Kruszewski, Denis Paperno, Raffaella Bernardi, and Marco Baroni. 2017. There is no logical negation here, but there are alternatives: Modeling conversational negation with distributional semantics. *Computational Linguistics*.

Alessandro Lenci. 2008. Distributional semantics in linguistic and cognitive research. *Italian journal of linguistics*, 20(1):1–31.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *Proceedings of 2013 International Conference on Learning Representations (ILCR)*.

Saif M Mohammad, Bonnie J Dorr, Graeme Hirst, and Peter D Turney. 2013. Computing lexical contrast. *Computational Linguistics*, 39(3):555–590.

Roser Morante and Caroline Sporleder. 2012. Modality and negation: An introduction to the special issue. *Computational Linguistics*, 38(2):223–260.

Lynne Murphy. 2003. *Semantic relations and the lexicon: Antonymy, synonymy and other paradigms*. Cambridge University Press.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2016. Integrating distributional lexical contrast into word embeddings for antonym-synonym distinction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 454–459.

Carita Paradis and Caroline Willners. 2006. Antonymy and negation—the boundedness hypothesis. *Journal of pragmatics*, 38(7):1051–1080.

Laura Rimell, Amandla Mabona, Luana Bulat, and Douwe Kiela. 2017. Learning to negate adjectives with bilinear models. In *Proceedings of the 15th Annual Meeting of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 71–78.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1201–1211.

Richard Socher, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D Manning, Andrew Y. Ng, Christopher Potts, et al. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642.

Michael Henry Tessler and Michael Franke. 2018. Not unreasonable: Carving vague dimensions with contraries and contradictions. In *Proceedings of the 40th Annual Meeting of the Cognitive Science Society*.

Nghia The Pham, Angeliki Lazaridou, and Marco Baroni. 2015. A multitask objective to inject lexical contrast into distributional semantics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and The 7th International Joint Conference of the Asian Federation of Natural Language Processing (ACL-IJCNLP)*, pages 21–26.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.

Chantal van Son, Emiel van Miltenburg, and Roser Morante Vallejo. 2016. Building a dictionary of affixal negations. In *Proceedings of the Workshop on Extra-Propositional Aspects of Meaning in Computational Linguistics*.

Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. 2010. A survey on the role of negation in sentiment analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing (NeSP-NLP)*, pages 60–68.

Bryan Wilkinson and Oates Tim. 2016. A gold standard for scalar adjectives. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC)*.

PRET: Prerequisite-Enriched Terminology. A Case Study on Educational Texts

Chiara Alzetta, Forsina Koceva, Samuele Passalacqua, Ilaria Torre, Giovanni Adorni
DIBRIS, University of Genoa (Italy)
{chiara.alzetta,frosina.koceva}@edu.unige.it,
samuele.passalacqua@dibris.unige.it,
{ilaria.torre,adorni}@unige.it

Abstract

English. In this paper we present PRET, a gold dataset annotated for prerequisite relations between educational concepts extracted from a computer science textbook, and we describe the language and domain independent approach for the creation of the resource. Additionally, we have created an annotation tool to support, validate and analyze the annotation.

Italiano. *In questo articolo presentiamo PRET, un dataset annotato manualmente rispetto alla relazione di prerequisito fra concetti estratti da un manuale di informatica, e descriviamo la metodologia, indipendente da lingua e dominio, usata per la creazione della risorsa. Per favorire l'annotazione, abbiamo creato uno strumento per il supporto, la validazione e l'analisi dell'annotazione.*

1 Introduction

Educational Concept Maps (ECM) are acyclic graphs which formally represent a domain's knowledge and make explicit the pedagogical dependency relations between concepts (Adorni and Koceva, 2016). A concept, in an ECM, is an atomic piece of knowledge of the subject domain. From a pedagogical point of view, the most important dependency relation between concepts is the prerequisite relation, that explicits which concepts a student has to learn before moving to the next. Several approaches have been proposed to extract prerequisite relations from various educational sources (Vuong et al., 2011; Yang et al., 2015; Gordon et al., 2016; Wang et al., 2016; Liang et al., 2017; Liang et al., 2018; Adorni et al., 2018). Textbooks in particular are a valuable resource for this task since they are designed to support the learning process respecting the prerequisite relation.

In the literature, the evaluation of the extracted prerequisite relations is usually performed through comparison with a gold standard produced by human subjects that annotate relations between concepts (see, among the others, (Talukdar and Cohen, 2012; Liang et al., 2015; Fabbri et al., 2018)). However, most of the evaluations lack a systematic approach or simply lack the details that allow them to be repeated. In this paper, we present our experience in building PRET (Prerequisite-Enriched Terminology), a gold dataset annotated with the prerequisite relation between pairs of concepts. The issues emerged with PRET led us to define a methodology and a tool for manual prerequisite annotation. The goal of the tool is to support the creation of gold datasets for validating automatic extraction of prerequisites. Both the PRET dataset and the tool are available online[1].

PRET was constructed in two main steps: first we exploited computational linguistics methods to extract relevant terms from a textbook[2], then we asked humans to manually identify and annotate the prerequisite relations between educational concepts. Since the terminology creation step was extensively described in Adorni et al. (2018), this paper mainly focuses on the annotation phase.

The annotation task consists in making explicit the prerequisite relations between two distinct concepts if the relation is somehow inferable from the text in question. We represent a concept as a domain–specific term denoting domain entities expressed by either single nominal terms (e.g. *internet, network, software*) or complex nominal structures with modifiers (e.g. *malicious software, trojan horse, HyperText Document*). Figure 1 shows

[1] http://teldh.dibris.unige.it/pret

[2] For the annotation we used chapter 4 of the computer science textbook "**Computer Science: An Overview**" (Brookshear and Brylow, 2015).

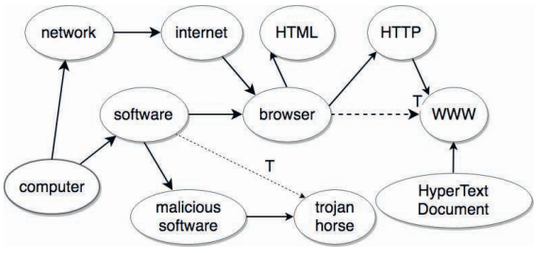


Figure 1: Sample of PRET represented as an ECM.

a sample of the ECM resulting from PRET. According to PRET dataset, an example of prerequisite relation is *network is a prerequisite of internet*, since a student has to know *network* before learning *internet*.

The paper is organized as follows. The related work pertaining to the proposed method is discussed in Section 2. Section 3 describes the methodology used for the creation of the PRET dataset and Section 4 presents the characteristics of the obtained gold dataset and the agreement computed for each pair of annotators together with other statistics about the data. Section 5 describes the main features of the annotation tool we designed. Section 6 concludes the paper.

2 Related Work

Automatic prerequisite identification is a task that gained growing interest in recent years, especially among scholars interested in automatic synthesis of study plans (Gasparetti et al., 2015; Yang et al., 2015; Agrawal et al., 2016; Alsaad et al., 2018). When applying automatic prerequisite extraction methods, a baseline for evaluation is needed. Despite being time consuming, creating manually annotated datasets is more effective and produces gold resources, which are still rare.

To the best of our knowledge, Talukdar and Cohen (2012) is the only case where crowd–sourcing is employed for annotation: they infer prerequisite relationship between concepts by exploiting hyper-links in Wikipedia pages and use crowdsourcing to validate those relations in order to have a gold training dataset for a classifier.

More frequently the annotation of prerequisite relations is performed by domain experts (Liang et al., 2015; Liang et al., 2018; Fabbri et al., 2018) or by students with a certain competence on the domain (Wang et al., 2015; Pan et al., 2017). When annotation is performed by non–experts, agreement usually results very low, so an expert can be consulted (Chaplot et al., 2016; Gordon et al., 2016). Regardless of the annotation methodology, we observe that in the mentioned related works prerequisite relation properties (i.e. irreflexivity, anti–symmetry, etc.) are rarely taken into account in the annotation instructions for annotators. For example, the fact that a concept cannot be annotated as prerequisite of itself is usually left unspecified.

To support the annotation of prerequisites between pairs of concepts, Gordon et al. (2016) developed an interface showing, for each concept of the domain, the list of relevant terms and documents. Although this can be of some support for the annotation providing certain useful information, it cannot be considered an annotation tool itself. According to our knowledge, a tool specifically designed for prerequisite structure annotation which also features agreement metrics is still missing.

3 Annotation Methodology

In Section 4 we will describe the PRET dataset, while here we present the annotation methodology that we used to build PRET and that we refined on the basis of such experience.

Concept identification. Our methodology for prerequisite annotation requires that concepts are extracted from educational materials, that we broadly define Document (D), and provided to annotators. Although we are conscious that a concept, as mental structure, might entail multiple terms, we simplify the problem of concept identification assuming that each relevant term of D represents a concept (Novak and Cañas, 2006). Thus, our list of concepts is a terminology (T) of domain–specific terms (either single or complex nominal structures) ordered according to the first appearance of the terms of T in D and where each concept corresponds to a single term.

For the task of prerequisite annotation, it does not matter if concepts are extracted automatically, manually or semi–automatically. To build PRET, we extracted concepts automatically. To identify our terminology T, we relied on Text-To-Knowledge ($T2K^2$) (Dell'Orletta et al., 2014), a software platform developed at the Institute of Computational Linguistics A. Zampolli of the CNR in Pisa. $T2K^2$ exploits Natural Language Processing, statistical text analysis and machine

learning to extract and organize the domain knowledge from a linguistically annotated text.

We applied T2K^2 to a text of 20,378 tokens distributed over 751 sentences. 185 terms were recognized as concepts of the domain (around 20% of the total number of nouns in the corpus). As expected, the extracted terminology contained both single nominal structures, such as *computer, network* and *software*, and complex nominal structures with modifiers, like *hypertext transfer protocol, world wide web* and *hypertext markup language*. The set of concepts did not go through any post–processing phase.

Annotators selection. The role of annotators is fundamental in order to obtain a gold dataset that represents the pedagogical relations expressed in the educational material. Consequently, the choice of annotators is crucial. As mentioned above, in the literature annotators are often domain experts (Liang et al., 2015; Liang et al., 2018; Fabbri et al., 2018) or students with some knowledge in that domain (Wang et al., 2015; Pan et al., 2017). Based on our experience with different types of annotators, we suggest that annotators should have enough knowledge to understand the content of the educational material. Otherwise, the annotation can be distorted by wrong comprehension of the relations between concepts. On the other hand, experts should not rely on their background knowledge to identify relations, since the goal of the annotation is to capture the knowledge embodied in the educational resource. To build PRET we recruited 6 annotators among professors and PhD students working in fields related to computer science, but eventually 2 of them revealed not to have enough knowledge for the task.

Annotation task. A prerequisite relation between two concepts A and B is defined as a dependency relation which represents what a learner must know/study (concept A), before approaching concept B. Thus, by definition, the prerequisite relation has the following properties: i) asymmetry: if concept A is a prerequisite of concept B, the opposite cannot be true (e.g. *network* is prerequisite of *internet*, so *internet* cannot be prerequisite of *network*); ii) irreflexivity: a concept cannot be prerequisite of itself; iii) transitiveness: if concept A is a prerequisite of concept B, and concept B of concept C, then concept A is also a prerequisite of concept C (e.g. *browser* is prerequisite of *HTTP*, *HTTP* is prerequisite of *WWW*, hence *browser* is prerequisite of *WWW* according to the transitive property).

To keep the annotation as uniform as possible, we provided the annotators with suggestions on how to perform the task together with the book chapter and the terminology extracted from it. Considering the material supplied, we asked annotators to trust the text considering only pairs of distinct concepts of T and annotating the existence of a prerequisite relation between the two concepts only if derivable from D. In our method, annotators should read the text and, for each new concept (i.e. never mentioned in the previous lines), identify all its prerequisites, but, if no prerequisite can be identified, they should not enter any annotation. We also wanted pedagogical relation properties to be preserved, so we asked to respect the irreflexive property not annotating self–prerequisites and to avoid adding transitive relations. Considering the topology of an ECM, we also asked annotators not to enter cycles in the annotation because they represent conceptually wrong relations. To better understand this point, consider the ECM in Figure 1: having a prerequisite relation between *computer* and *network* and between *network* and *internet*, entering a relation where *internet* is prerequisite of *computer* would create a cycle (loop).

The output of the annotation of each annotator is an *enriched terminology*: a set of concepts paired and enhanced with the prerequisite relation. The enriched terminology can be used to create an ECM where each concept is a node and the edges are prerequisite relations identified by humans (see Figure 1).

Annotation validation. Human annotators are not immune from making mistakes and violating the supplied recommendations. The tool we propose addresses this issue by introducing controls to prevent the annotators from making errors (e.g. cycles, reflexive relations, symmetric relations). In the next section we will describe the approach we used to identify some mistakes by using graph analysis algorithms.

Annotators agreement evaluation. Our experience and the literature (Fabbri et al., 2018) show that human judgments about prerequisite identification can vary considerably, even when strict guidelines are provided. This can depend on several factors, including the subjectivity of annotators and the type and complexity of D. Evaluating the annotators' agreement can be useful to assess

Relation Type	Weight	Count (%)
Non–prerequisite	0	33,699 (98.46%)
Prerequisite	All weights	526 (154%)
1 annot.	0.25	293 (55.70%)
2 annot.	0.50	131 (24.90%)
3 annot.	0.75	75 (14.26%)
4 annot.	1	27 (5.13%)
Total number of pairs		34,225

Table 1: Relations and weight distribution in PRET dataset.

if the gold dataset is to be trusted or further annotators are required. Section 4 will describe the measures we used to evaluate annotators' agreement in PRET.

The final combination of the enriched terminologies produced by each annotator is a necessary step to build a gold dataset but, due to space constraints, below we will only present our approach, while a survey on combination metrics is out of the scope of this paper.

4 The PRET Dataset

The PRET gold dataset consists of 34,225 concept pairs obtained by all possible combinations of the elements in the concepts set (excluding self–prerequisites). Pairs vary with respect to the relation weight, computed for each pair by dividing the number of annotators that annotated the pair by the total number of annotators. Only 1.54% (526) of the pairs has a relation weight higher than 0 (i.e. it was annotated as prerequisite by at least one annotator). Details about the distribution of prerequisite relations and respective weights are reported in Table 1.

55.70% (293) of the prerequisite pairs was identified by only one annotator, meaning that it is hard for humans to agree on what a prerequisite is. We further investigate this aspect in section 4.1.

The analysis of the dataset carried out before applying validation checks highlighted some critical issues: some transitive relations were explicitly annotated and some cycles were erroneously added in the dataset, violating the instructions. While cycles are due to distraction, transitive relations are hard to recognize per se, especially when broad terms are involved (e.g. *computer, software, machine*).

In order to study how these issues impact the dataset, each annotation was validated against cycles and transitive relations obtaining 5 dataset variations, in addition to the original annotation. The validation was conducted on the ECM derived from the enriched terminology of each annotator using a graph analysis algorithm. We operated on cycles and transitive relations. In some variations, the latter were added if the pair of concepts in the ECM is connected by a path shorter than a certain threshold, defined by considering the ECM diameter, while cycles were either preserved or removed depending on the variation we wanted to obtain.

Eventually, we obtained the following annotation variations: *no cycles* (removing cycles), *cycles and transitive* (preserving cycles and adding transitive relations), *cycles and non–transitive* (preserving cycles and keeping only direct links), *no cycles and transitive* (removing cycles and adding transitivity) and *no cycles and non–transitive* (removing both cycles and transitivity).

4.1 Annotators Agreement in PRET

Following Artstein and Poesio (2008), we computed the agreement between multiple annotators using Fleiss' k (Fleiss, 1971) and between pairs of annotators using Cohen's k (Cohen, 1960). Using the scale defined by Landis and Koch (1977), Fleiss' k values show *fair agreement*, suggesting that prerequisite annotation is difficult. Similar tasks obtained comparable or lower values, confirming our hypothesis: Gordon et al. (2016) measured the agreement as Pearson Correlation obtaining 36%, while Fabbri et al. (2018) and Chaplot et al. (2016) obtained respectively 30% and 19% of Fleiss' k.

Compared to the other variations, removing cycles and adding transitive relations showed the highest improvement on the agreement, also for pairs of annotators (Table 2). Our results suggest that different competence level entails different annotations and values of agreement, confirming previous results (Gordon et al., 2016): lower agreement can be observed when annotator 4 (quasi–expert) is involved, possibly due to the lower competence level if compared to the other annotators. Annotator 4 is also the one who considered the highest number of transitive relations, producing a more connected ECM: it is likely that when the competence in the domain is lower, a person tends to consider a higher number of prerequisites for each concept. On the other hand, annotators with more experience show even *moderate* (pairs A1-A3 and A2-A3) or *substantial agree-*

Metric		Orig.	No Cycl. & Trans.	Diff
Fleiss's k	All raters	38.50%	**39.94%**	+1.44
Cohen's k	A1-A2	34.46%	42.81%	**+8.35**
	A1-A3	57.80%	50.84%	**-6.96**
	A1-A4	37.59%	39.29%	+1.70
	A2-A3	56.50%	63.62%	+7.12
	A2-A4	28.02%	29.42%	+1.40
	A3-A4	25.35%	25.71%	+0.36

Table 2: Agreement values and differences for two annotation variations.

ment (pair A2-A3 for the variation). Adding transitive relations and removing cycles generally improves the agreement values also when we consider pairs: we notice an increase of 8.35 points for A1-A2. The only exception is observed for the pair A1-A3, which experienced a decrease of almost 7 points. The cause is though to be the number of transitive relations considered by annotator 3, which is around one third of the transitive relations annotated by annotator 1: the validation creates more distance between the two annotations reducing the agreement.

As a support for the annotation, the experts used a $n \times n$ matrix of the terminology T where they entered a binary value in the intersection between two concepts to indicate the presence of a prerequisite relation. We believe that our results are partially influenced by the instrument we used to perform the annotation: a large matrix structure is likely to cause distraction errors and does not perform validation checks during the annotation. Based on this experience and the encountered issues, we developed an annotation tool able to support and validate the annotation. It will be described in the next section.

5 Annotation and Analysis Tool

We provide a language and domain independent prototype tool which aims on the one hand to support and validate the annotation process and on the other hand to perform annotation analysis. All its main features have been designed taking into account real problems encountered while building PRET. Thus, this tool is highly valuable for annotators because specifically addresses annotators' needs and, at the same time, avoids possible annotation biases. In particular, the tool has three main functionalities: annotation support, annotation representation and analysis of the results.

To support the annotation, the user is provided with the terminology T as a list L of concepts ordered by their first occurrence in the text. This is done in order to give the annotator an overview of the context in which the concept occurs. We observed that the textual context plays a crucial role in deciding which concepts are prerequisites of the one under observation, so for each term we show the list of other terms with visual indication of the progress in the text. Additionally, as said before, the tool validates the map resulting from the annotation against the existence of symmetric relations, transitivity and cycles.

Once the annotation is completed, the user can choose to generate different types of visualization of her/his annotation. The goal of this functionality is to provide information visualization and data summarization for analyzing and exploring the result of the annotation. We provide the following different views: Matrix (ordered by concept frequency, clusters, temporal, occurrence or alphabetic order), Arc Diagram, Graph and Clusters. Furthermore, the Data Synthesis task provides the number of concepts, number of relations, number and list of disconnected nodes and transitive relations.

Lastly, the tool computes the agreement between relations inserted by all annotators who took part in the task (see Section 4.1) and provides visualization of the final dataset, which results as a combination of all users' annotation. This feature also outputs a Data Synthesis that provides the number of relations of every annotator, number of transitive relations and the direction of conflicting relations between annotators.

The demo version of the tool is available online at the URL provided in the Introduction.

6 Conclusion and Future Work

In this paper, we described PRET, a gold dataset manually annotated for prerequisite relations between pairs of concepts; moreover we presented the methodology we adopted and a tool to support prerequisite annotation. The case study, even limited as for the number of annotators and the educational material, was a reasonably good training ground to set the basis to define a methodology for prerequisite annotation and to identify the major issues related to this task. Moreover, the analysis of the annotation provided insights for automatic identification of concepts and prerequisites, that will be investigated in future work.

References

Giovanni Adorni and Frosina Koceva. 2016. Educational concept maps for personalized learning path generation. In *Conference of the Italian Association for Artificial Intelligence*, pages 135–148. Springer.

Giovanni Adorni, Felice Dell'Orletta, Frosina Koceva, Ilaria Torre, and Giulia Venturi. 2018. Extracting dependency relations from digital learning content. In *Italian Research Conference on Digital Libraries*, pages 114–119. Springer.

Rakesh Agrawal, Behzad Golshan, and Evangelos Papalexakis. 2016. Toward data-driven design of educational courses: a feasibility study. *Journal of Educational Data Mining*, 8(1):1–21.

Fareedah Alsaad, Assma Boughoula, Chase Geigle, Hari Sundaram, and Chengxiang Zhai. 2018. Mining MOOC lecture transcripts to construct concept dependency graphs. In *Proceedings of the 11th International Conference on Educational Data Mining, EDM 2018, Buffalo, NY, USA, July 15-18, 2018*.

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Glenn Brookshear and Dennis Brylow, 2015. *Computer Science: An Overview, Global Edition*, chapter 4 Networking and the Internet. Pearson Education Limited.

Devendra Singh Chaplot, Yiming Yang, Jaime G. Carbonell, and Kenneth R. Koedinger. 2016. Data-driven automated induction of prerequisite structure graphs. In *Proceedings of the 9th International Conference on Educational Data Mining, EDM 2016, Raleigh, North Carolina, USA, June 29 - July 2, 2016*, pages 318–323.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Felice Dell'Orletta, Giulia Venturi, Andrea Cimino, and Simonetta Montemagni. 2014. T2k^2: a system for automatically extracting and organizing knowledge from texts. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.

Alexander R Fabbri, Irene Li, Prawat Trairatvorakul, Yijiao He, Wei Tai Ting, Robert Tung, Caitlin Westerfield, and Dragomir R Radev. 2018. Tutorialbank: A manually-collected corpus for prerequisite chains, survey extraction and resource recommendation. In *ACL*.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Fabio Gasparetti, Carla Limongelli, and Filippo Sciarrone. 2015. Exploiting wikipedia for discovering prerequisite relationships among learning objects. In *2015 International Conference on Information Technology Based Higher Education and Training, ITHET 2015, Lisbon, Portugal, June 11-13, 2015*, pages 1–6.

Jonathan Gordon, Linhong Zhu, Aram Galstyan, Prem Natarajan, and Gully Burns. 2016. Modeling concept dependencies in a scientific corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 866–875.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Chen Liang, Zhaohui Wu, Wenyi Huang, and C Lee Giles. 2015. Measuring prerequisite relations among concepts. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1668–1674.

Chen Liang, Jianbo Ye, Zhaohui Wu, Bart Pursel, and C Lee Giles. 2017. Recovering concept prerequisite relations from university course dependencies. In *AAAI*, pages 4786–4791.

Chen Liang, Jianbo Ye, Shuting Wang, Bart Pursel, and C Lee Giles. 2018. Investigating active learning for concept prerequisite learning. *Proc. EAAI*.

Joseph D. Novak and Alberto J. Cañas. 2006. The theory underlying concept maps and how to construct and use them. research report 2006-01 Rev 2008-01, Florida Institute for Human and Machine Cognition.

Liangming Pan, Chengjiang Li, Juanzi Li, and Jie Tang. 2017. Prerequisite relation learning for concepts in moocs. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1447–1456.

Partha Pratim Talukdar and William W Cohen. 2012. Crowdsourced comprehension: predicting prerequisite structure in wikipedia. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 307–315. Association for Computational Linguistics.

Annalies Vuong, Tristan Nixon, and Brendon Towle. 2011. A method for finding prerequisites within a curriculum. In *Proceedings of the 4th International Conference on Educational Data Mining, Eindhoven, The Netherlands, July 6-8, 2011*, pages 211–216.

Shuting Wang, Chen Liang, Zhaohui Wu, Kyle Williams, Bart Pursel, Benjamin Brautigam, Sherwyn Saul, Hannah Williams, Kyle Bowen, and C Lee Giles. 2015. Concept hierarchy extraction from textbooks. In *Proceedings of the 2015 ACM Symposium on Document Engineering*, pages 147–156. ACM.

Shuting Wang, Alexander Ororbia, Zhaohui Wu, Kyle Williams, Chen Liang, Bart Pursel, and C Lee Giles. 2016. Using prerequisites to extract concept maps fromtextbooks. In *Proceedings of the 25th acm international on conference on information and knowledge management*, pages 317–326. ACM.

Yiming Yang, Hanxiao Liu, Jaime Carbonell, and Wanli Ma. 2015. Concept graph learning from educational data. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 159–168. ACM.

Distributional Analysis of Verbal Neologisms: Task Definition and Dataset Construction

Matteo Amore
University of Pavia / Pavia, Italy
CELI Language Technology / Turin, Italy
matteo.amore01@universitadipavia.it

Stephen McGregor
LATTICE - CNRS & École normale supérieure / Montrouge, France
Université Sorbonne nouvelle Paris 3 / Paris, France
semcgregor@hotmail.com

Elisabetta Jezek
University of Pavia / Pavia, Italy
Department of Humanities
jezek@unipv.it

1 Abstract

English In this paper we introduce the task of interpreting verbal neologism (VNeo) for the Italian language making use of a highly context-sensitive distributional semantic model (DSM). The task is commonly performed manually by lexicographers verifying the contexts in which the VNeo appear. Developing such a task is likely to be of use from a cognitive, social and linguistic perspective. In the following, we first outline the motivation for our study and our goal, then focus on the construction of the dataset and the definition of the task.

Italian *In questo contributo introduciamo un task di interpretazione dei neologismi verbali (Vneo) in italiano, utilizzando un modello di semantica distribuzionale altamente sensibile al contesto. Questa attività è comunemente svolta manualmente dai lessicografi, i quali verificano il contesto in cui il Vneo appare. Sviluppare questo tipo di task può rivelarsi utile da una prospettiva linguistica, cognitiva e sociale. Di seguito presenteremo inizialmente le motivazioni e gli scopi dell'analisi, concentrandoci poi sulla costruzione del dataset e sulla definizione del task.*

1 Introduction: motivation and goals

Studying neologisms can tell us several things. From a lexicographic point of view, neologisms can show trends that a language is following. In our opinion, they can also shed light on various aspects related to linguistic creativity; when speakers use new words (coined by themselves, or recently coined by someone else), they expect that the hearer can understand what they have just said.[1] Reversing the perspective, from the point of view of the hearers, when they encounter a word for the first time, they are generally capable of making hypotheses about the meaning of that word. The process of understanding unknown words involves the employment of previously acquired information. This knowledge can come from various sources: experience of the world, education, and contextual elements;[2] in this contribution we focus on linguistic contextual (namely co-occurrence) information.

For computational linguistics, neologisms raise some intriguing issues: automatic detection (especially for languages which do not separate written words with blank spaces); lemmatisation; POS tagging; semantic analysis; and so forth.

In this paper we present the task we have developed in order to interpret neologisms, using a context-sensitive DSM described by McGregor *et al.* (McGregor et al., 2015). This model was built to represent concepts in a spatial configuration, making use of a computational technique that creates conceptual subspaces. With the help of this DSM we intend to analyse the behaviour of a sub-group of neologisms, namely verbal neologisms (see Amore 2017 for more background).

Our goal is primarily linguistic. We intend to investigate the interpretation of VNeo, measuring the semantic salience of candidate synonyms by way of geometries indicated by an analysis of co-occurrence observations of VNeos. For instance, we expect that the VNeo *googlare* 'to google' and a verb like *cercare* 'to search' are geometrically related in a *subspace* specific to the conceptual context of the neologism.

[1] This is not the case of neologisms created for advertising, brand names or marketing purposes in general (Lehrer, 2003:380).

[2] All of these aspects are investigated, for example, in the field of Contextual Vocabulary Acquisition (Rapaport & Ehrlich, 2000).

The interpretation of neologisms presents two main challenges: a) analysing verbs using vectors built only upon co-occurrences (thus excluding argument structures) is notoriously a difficult task for DSM;[3] b) neologisms are, by definition, words whose frequency is (very) low, because their use is (still) not widespread. Thus, it represents a challenge for DSM models exactly because the vectors for most VNeo will rely upon few occurrences. In order to evaluate our results, we will compare them with the ones obtained using the Word2Vec model (Mikolov et al., 2013a), and with a gold standard consisting in human judgments on semantic relatedness (synonymy). The paper is structured as follows. In section 2 we introduce the DSM model that we employ in our task, and in section 3 we describe the construction of VNeo dataset and the problems we encountered. Finally, in section 4 we outline the task and present some preliminary thoughts on expected results.

2 Distributional Semantic Modelling

DSM is a technique for building up measurable, computationally tractable lexical semantic representations based on observations of the way that words co-occur with one another across large-scale corpora. This methodology is grounded in the *distributional hypothesis,* which maintains that words that are observed to have similar co-occurrence profiles are likely to be semantically related (Harris, 1954; Sahlgren, 2008). In general, a DSM consists of a high-dimensional vector space in which words correspond to vectors, and the geometric relationship between vectors is expected to indicate something about the semantic relationship between the associated words. The relationship most typically modelled is general semantic *relatedness,* as opposed to more precise indications of, for instance, *similarity* (Hill et al., 2015), but distributional semantic models have been effectively applied to tasks ranging from language modelling (Bengio, 2009) to metaphor classification (Gutiérrez et al., 2016) and the extrapolation of more fine-grained intensional correspondences between concepts (Derrac and Schockaert, 2015).

Standard DSM techniques present two problems for the task of interpreting neologisms. First, distributional representations are predicated on many observations of a word across a large-scale corpus: it is the plurality of context which gives these representations their semantic nuance. Second, the spaces generated by standard approaches like matrix factorisation and neural networks are abstract, in the sense that their dimensions are not interpretable; as such, typical distributional semantic models are not sensitive to the context specific way in which meaning arises in the course of language use. McGregor et al. (2015) have proposed a *context-sensitive* approach to distributional semantic modelling that seeks to overcome this second problem by using contextual information to project semantic representations into lower dimensional conceptual perspectives in an on-line way.

This methodology entails the selection of sets of dimensions from a base space of co-occurrence statistics that are in some sense conceptually *salient* to the context being modelled. The selection of salient features facilitates the projection of subspaces in which the geometric situation of and relationship between word-vectors are expected to map to a specific conceptual context. This technique has been applied to tasks involving context sensitive semantic phenomena such as metaphor rating (Agres et al., 2016), analogy completion (McGregor et al., 2016), and the classification of semantic type coercion (McGregor et al., 2017).

With regard to the first problem of data sparsity, we propose that the facility of the dynamically contextual approach for handling the *ad hoc* emergence of concepts (Barsalou, 1993) should provide a way of mapping from relatively few observations of neologisms, possibly taken outside the data used to build the underlying model, to context specific perspectives on distributional semantic representations.

3 Verbal Neologisms: dataset, corpus and lemmatisation

We will now explain the methodology we use in our analysis, and describe the resources we exploit highlighting their main features.

3.1 Sources for the neologisms list

To select the VNeo to be analysed, we extract data from pre-existing lists of Italian neologisms. These lists come from three websites: a)

[3] Cf. Bundell et al., 2017 and Chersoni et al., 2016.

treccani.it[4] b) iliesi.cnr.it/ONLI/[5] c) accademiadellacrusca.it.[6] (a) and (b) are manually compiled and validated: they contain words manually found in some widely read newspapers but not (yet) included in Italian dictionaries, coherently with the lexicographical definition of neologisms (cf. Adamo & Della Valle 2017). (c) consists of a list of words that, according to the users of the website, should be included in dictionaries. There is no curating of these suggestions (except the removal of swearwords); thus some neologisms might already be included in dictionaries. We chose to use this list because it allows analysing words which are perceived as new from a community of Italian speakers. In this way we intend to highlight the perspective of the hearers encountering new words.

Within the lists, we select only the verbs, obtaining a set of 504 VNeo. Of these VNeo, we check their presence in the itTenTen16 corpus, which we will also use to create the distributional vector space. 340 VNeo are attested in the corpus: 108 have between 10 and 99 occurrences; 79 between 100 and 999 occurrences; and 26 have more than 1000 occurrences.

Instead of using heuristic techniques that might have identified neologisms within the corpus (e.g. computing less frequent words and manually checking their presence in dictionaries),[7] we chose to rely on lists because we intend to study words whose use is wider and not restricted only to the web domain.

3.3 itTenTen16 corpus

We conduct an analysis of the itTenTen16 corpus (Jakubíček et al. 2013) because it is the most up-to-date corpus available for Italian. It is also a web-based corpus, and so particularly well fitted to examine neologisms: in fact, the web and IT domain is a notable source of new words and, especially, of new loanwords. As the corpus dimensions are sizeable (4.9 billion tokens), we will use a random sample of the full corpus for purposes of computability. This sample will correspond to ⅕ of the original corpus.

Starting from the corpus, the base DSM is built based on observations of the most frequent 200,000 words (defined as *vocabulary*) and their contextual information, considering a co-occurrence window of 5 words on either side of a target word. For the purposes of this study, we consider the VNeos included in the vocabulary. In this way we obtain the base space.

In order to project a subspace contextualised by a VNeo, we consider the co-occurrence features with the highest mutual information statistics associate with that particular VNeo. So, for instance, we find the following salient features:

customizzare 'to customise' [city; modellazione; illustrato; type; batch; editare; nastro; segmentare; preferenza; iconico; ...]

resettare 'to reset' [reset; password; formattare; bios; clempad; clementoni; fonera; resettare; centralina; router; ...]

googlare 'to google' [telespettatore; pdf; tecnologia; informazione; addirittura; vi; chiave; invito; risposta; sapere; ...].

These features are associated with the maximum mutual information values in terms of their co-occurrence with each of the corresponding input neologisms.

Some other VNeos represented in the vocabulary are: *postare* ‘to post’, *taggare* ‘to tag’, *twittare* ‘to tweet’, *spammare* ‘to spam’, *attenzionare* ‘to warn’, *spoilerare* ‘share information that reveals plot of a book or film’, *bloggare* ‘to blog’, *loggare* ‘to log’, *switchare* ‘to switch’.

It is worth noting that we create vectors starting from lemmas (not tokens). Our analysis highlighted the presence of some inaccuracies in the automatic lemmatisation of neologisms,[8] which was already present in the original corpus.[9] In a future investigation we are planning to compare the results produced with the original lemmatised corpus against the results obtained from a corpus version, where the lemmatisation will be corrected. This correction process might be performed using regular expressions, in order

[4] http://www.treccani.it/magazine/lingua_italiana/neologismi (last consulted 10/04/2018)

[5] http://www.iliesi.cnr.it/ONLI/BD.php (last consulted 02/05/2018)

[6] http://www.accademiadellacrusca.it/it/lingua-italiana/parole-nuove (last consulted 02/05/2018)

[7] We are aware that this might correspond to the loss of some other neologisms contained in the corpus.

[8] Neologisms are not stored in common word-lists, and they are (usually) rare words, thus presenting difficulties for machine learning techniques.

[9] The lemmatisation is obtained using the TreeTagger tool (Schmid, 1994) with Baroni’s parameter file (http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/)

to capture specific VNeos token.[10]

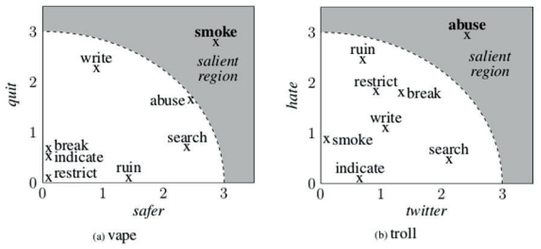


Figure 1: Two subspaces projected based on two co-occurrence dimensions closely associated with the words (a) *vaped* and *vaping*, and (b) *trolled* and *trolling*, as observed in a small set of recent posts on Twitter. Among vectors for a number of candidate interpretations of neologisms, we see appropriate interpretations emerging based on distance from the origin in each contextualised subspace, based on PMI statistics extrapolated from co-occurrences observed across English language Wikipedia.

4. Interpreting VNeo using geometrical subspaces

As referenced in §1, our goal is to verify whether the meaning of a neologism can be induced from its context through distributional techniques, in particular by discovering verbs with salient geometric features in a contextualised subspace.

To this end, we organize the task as follows. Starting from a subset of the most frequent VNeos found in the corpus (§3), we first build subspaces for VNeos using the DSM model presented in §2. Subspaces are created by selecting the sets of dimensions that are conceptually salient to the context being modelled: each dimension in a subspace corresponds to a specific co-occurrence feature (i.e. a word). By finding a whole set of co-occurrences and using these to generate a relatively high-dimensional projection, we hope to establish a general contextualised conceptual profile and to overcome the peculiarities associated with low-frequency targets. For example, if the model finds that *googlare* 'to google' co-occurs with words like *nome* 'name', *indirizzo* 'address', and *sito* 'website', we use those co-occurrences as a basis for a projection of a *subspace* in which one could predict to find terms like *cercare* 'search' using geometric techniques.

Context can be defined in an open ended way in these models. For instance, the salient co-occurrence features of a single word can be used to generate a subspace. Small sets of words, either components of observed compositions (McGregor et al., 2017) or groups of conceptually related terms (McGregor et al., 2015) have also been used to generate semantically productive subspaces. In the small example illustrated in Figure 1, on the other hand, dimensions are defined explicitly in terms of the salient words associated with a small number of very recent observations of two different neologisms in use, specifically extrapolated from the salient co-occurrence features of Twitter posts in which the targeted neologisms are mentioned.

Contextualised subspaces can be explored in terms of the geometric features of word-vectors projected into those subspaces. So, for instance, McGregor et al. (2015) propose a norm method, by which word-vectors salient in a particular context will emerge as being far from the origin. This phenomenon is observed with appropriate interpretations percolating into the salient regions even in the low-dimensional toy examples illustrated in Figure 1, which involves a dynamically contextual DSM built from English language Wikipedia. Choices about context selection techniques, geometric characteristics of subspaces to be explored, and modelling parameters including dimensionality of projections will be the subject of our forthcoming experiments.

In order to evaluate the model, we will compare our results against the results obtained applying the Word2Vec model to the same corpus (Mikolov et al., 2013a).

With further investigations we will also test this model using a gold standard consisting of human judgments on VNeos interpretations collected for this purpose. Similarity judgments will be provided by two native speakers with significant background in linguistics. Specifically, the dataset will consist of verb pairs in which VNeo are grouped with more common verbs (*googlare* and *cercare*) based on human ratings collected in the form of a TOEFL-like multiple-choice synonymy test.[11]

[10] Regular expressions might be useful, within the corpus, to find an inflected form of a verb (lemmatised as it is) and replace it with the correct lemma: e.g. find lemma `googlav`. (meaning *googlavo, googlavi,* etc.*)* and replace it with *googlare*.

[11] Here the task is to determine, for a number of target words, the closest synonym from a choice of four alternatives.

4 Conclusion

The aim of the task presented here is to investigate the importance of linguistic context for the interpretation of neologisms, grounding the analysis in a context-sensitive DSM. With this task we intend to tackle issues connected with creativity processes and the environmental (contextual) sensibility typical of human cognition. In addition, we apply, for the first time, this DSM to Italian, providing a new semantic resource for the analysis of the language. Further studies may compare our results with other DSMs, and/or study what the semantic relations found with this specific approach reveal about other phenomena belonging to different linguistic levels (e.g. syntax).

References

Giovanni Adamo and Valeria Della Valle. 2017. *Che cos'è un neologismo?*. Carocci Editore, Roma.

Kat Agres, Stephen McGregor, Karolina Rataj, Matthew Purver, and Geraint A. Wiggins. 2016. Modeling metaphor perception with distributional semantics vector space models. In *Workshop on Computational Creativity, Concept Invention, and General Intelligence*, 08/2016.

Matteo Amore. 2017. I Verbi Neologici nell'Italiano del Web: Comportamento Sintattico e Selezione dell'Ausiliare. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017)*, Rome, Italy, December 11-13, 2017.

Lawrence W. Barsalou. 1993. Flexibility, structure, and linguistic vagary in concepts: Manifestations of a compositional system of perceptual symbols. In A.C. Collins, S.E. Gathercole, and M.A. Conway, editors, *Theories of memory*, pages 29–101. Lawrence Erlbaum Associates, London.

Yoshue Bengio. 2009. Learning deep architecture for AI. *Machine Learning*, 2(1):1–127.

Benjamin Blundell, Mehrnoosh Sadrzadeh, Elisabetta Jezek. 2017. Experimental Results on Exploiting Predicate-Argument Structure for Verb Similarity in Distributional Semantics. In Clasp Papers in Computational Linguistics, vol. 1, pages 99-106.

Emmanuele Chersoni, Enrico Santus, Alessandro Lenci, Philippe Blache, Chu-Ren Huang 2016. Representing Verbs with Rich Contexts: an Evaluation on Verb Similarity, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* Association for Computational Linguistics, pages 1967–1972.

Bob Coecke, Mehrnoosh Sadrzadeh, and Stephen Clark. 2011. Mathematical foundations for a compositional distributed model of meaning. *Linguistic Analysis,* 36:345–384.

Joaquín Derrac and Steven Schockaert. 2015. Inducing semantic relations from conceptual spaces: A data-driven approach to plausible reasoning. *Artificial Intelligence*, 228:66–94.

E. Darío Gutiérrez, Ekaterina Shutova, Tyler Marghetis, and Benjamin K. Bergen. 2016. Literal and metaphorical senses in compositional distributional semantic models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Zellig Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with genuine similarity estimation. *Computational Linguistics*, 41(4):665–695.

Miloš Jakubíček, Adam Kilgarriff, Vojtěch Kovář, Pavel Rychlỳ, and Vít Suchomel. 2013. The tenten corpus family. In *7th International Corpus Linguistics Conference CL*, pages 125–127.

Adrienne Lehrer. 2003. Understanding trendy neologisms. *Italian Journal of Linguistics*, 15:369–382.

Stephen McGregor, Kat Agres, Matthew Purver, and Geraint Wiggins. 2015. From distributional semantics to conceptual spaces: A novel computational method for concept creation. *Journal of Artificial General Intelligence*, 6(1):55–89.

Stephen McGregor, Matthew Purver, and Geraint Wiggins. 2016. Words, concepts, and the geometry of analogy. In *Proceedings of the Workshop on Semantic Spaces at the Intersection of NLP, Physics and Cognitive Science (SLPCS)*, pages 39–48.

Stephen McGregor, Elisabetta Jezek, Matthew Purver, and Geraint Wiggins. 2017. A geometric method for detecting semantic coercion. In *Proceedings of 12th International Workshop on Computational Semantics*.

Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean. 2013a. Efficient Estimation of Word Representations in Vector Space. In *ICLR Workshop Papers*.

Jeff Mitchell and Mirella Lapata. 2010. Composition in Distributional Models of Semantics. Cognitive Science 34:1388–1429.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word

representations. *Proceedings of NAACL-HLT 2018,* pages 2227–2237.

William J. Rapaport and Karen Ehrlich. 2000. A computational theory of vocabulary acquisition. In Stuart Charles Shapiro and Lucja M. Iwánska, editors, *Natural language processing and knowledge representation: language for knowledge and knowledge for language*. MIT Press, Cambridge, MA.

Magnus Sahlgren. 2008. The distributional hypothesis. *Italian Journal of Linguistics*, 20(1):33–54.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of International Conference on New Methods in Language Processing*. Manchester, UK.

Parsing Italian texts together is better than parsing them alone!

Oronzo Antonelli
DISI, University of Bologna, Italy
antonelli.oronzo@gmail.it

Fabio Tamburini
FICLIT, University of Bologna, Italy
fabio.tamburini@unibo.it

Abstract

English. In this paper we present a work aimed at testing the most advanced, state-of-the-art syntactic parsers based on deep neural networks (DNN) on Italian. We made a set of experiments by using the Universal Dependencies benchmarks and propose a new solution based on ensemble systems obtaining very good performances.

Italiano. *In questo contributo presentiamo alcuni esperimenti volti a verificare le prestazioni dei più avanzati parser sintattici sull'italiano utilizzando i* treebank *disponibili nell'ambito delle* Universal Dependencies*. Proponiamo inoltre un nuovo sistema basato sull'*ensemble parsing *che ha mostrato ottime prestazioni.*

1 Introduction

Syntactic parsing of morphologically rich languages like Italian often poses a number of hard challenges. Various works applied different kinds of freely available parsers on Italian training them using different resources and different methods for comparing their results (Lavelli, 2014; Alicante et al., 2015; Lavelli, 2016) and gather a clear picture of the syntactic parsing task performances for the Italian language. In this direction seems relevant to cite the EVALITA[1] periodic campaigns for the evaluation of constituency and dependency parsers devoted to the syntactic analysis of Italian (Bosco and Mazzei, 2011; Bosco et al., 2014).

Other studies regarding the syntactic parsing of Italian tried to enhance the parsing performances by building some kind of *ensemble systems* (Lavelli, 2013; Mazzei, 2015).

[1] http://www.evalita.it

By looking at the cited papers we can observe that they evaluated the state-of-the-art parsers before the "neural net revolution" not including the last improvements proposed by new research studies.

The goal of this paper is twofold: first, we would like to test the effectiveness of parsers based on the newly-proposed technologies, mainly deep neural networks, on Italian, and, second, we would like to propose an ensemble system able to further improve the neural parsers performances when parsing Italian texts.

2 The Neural Parsers

We considered nine state of the art parsers representing a wide range of contemporary approaches to dependency parsing whose architectures are based on neural network models (see Table 1). We set-up each parser using the data from the Italian Universal Dependencies (Nivre et al., 2016) treebank, UD Italian 2.1 (general texts) and UD Italian PoSTWITA 2.2 (tweets). For all parsers, we used the default settings for training, following the recommendation of the developers.

In Chen and Manning (2014) dense features are used to learn representations of words, tags and labels using a neural network classifier in order to take parsing decisions within a transition-based greedy model. To address some limitations, in Andor et al. (2016) the authors augmented the parser model with a beam search and a conditional random field loss objective. The work of Ballesteros et al. (2015) extends the parser defined in Dyer et al. (2015) introducing character-level representation of words using bidirectional LSTMs to improve the performance of *stack-LSTM* model which learn representations of the parser state. In Kiperwasser and Goldberg (2016) the bidirectional LSTMs recurrent output vector for each word is concatenated with any possible heads recurrent vector, and the result is used as input to a

multi-layer perceptron (MLP) network that scores each resulting edge. Cheng et al. (2016) propose a bidirectional attention model which uses two additional unidirectional RNN, called left-right and right-left query component. Based on Kiperwasser and Goldberg (2016) and Cheng et al. (2016) model, in Dozat and Manning (2017) a biaffine attention mechanism is used, instead of traditional MLP-based attention. The model proposed in Nguyen et al. (2017) train a neural network model that learn jointly POS tagging and graph-based dependency parsing. The model uses a bidirectional LSTM to learn POS tagging and the Kiperwasser and Goldberg (2016) approach for dependency parsing. Shi et al. (2017a,b) described a parser that combines three parsing paradigms using a dynamic programming approach.

Parser Ref.-Abbreviation	Method	Parsing
(Chen and Manning, 2014) - CM14	Tb: a-s	Greedy
(Ballesteros et al., 2015) - BA15	Tb: a-s	Be-se
(Kiperwasser and Goldberg, 2016)- KG16:T	Tb: a-h	Greedy
(Kiperwasser and Goldberg, 2016)- KG16:G	Gb: a-f	Eisner
(Andor et al., 2016) - AN16	Tb: a-s	Beam-S
(Cheng et al., 2016) - CH16	Gb: a-f	cle
(Dozat and Manning, 2017) - DM17	Gb: a-f	cle
(Shi et al., 2017a,b)- SH17	Tb: a-h./ -eager	Greedy
	Gb: a-f	Eisner
(Nguyen et al., 2017) - NG17	Gb: a-f	Eisner

Table 1: All the neural parsers considered in this study with their fundamental features as well as their abbreviations used throughout the paper. In this table "Tb/Gb" means "Transition/Graph-based", "Beam-S" means "Beam-search" and "a-s/h/f" means "arc-standard/hybrid/factored".

We trained, validated and tested the nine considered parsers, as well as all the proposed extensions, by considering three different setups:

- **setup0**: only the UD Italian 2.1 dataset;
- **setup1**: only the UD Italian PoSTWITA 2.2 dataset;
- **setup2**: UD Italian 2.1 dataset joined with the UD Italian PoSTWITA 2.2 dataset (train and validation sets) keeping the test set of PoSTWITA 2.2;

After the influential paper from Reimers and Gurevych (2017) it is clear to the community that reporting a single score for each DNN training session could be heavily affected by the system initialisation point and we should instead report the mean and standard deviation of various runs with the same setting in order to get a more accurate picture of the real systems performances and make more reliable comparisons between them.

Table 2 shows the parsers performances on the test set for the three setups described above executing the training/validation/test cycle for 5 times. In any setup the DM17 parser exhibits the best performances, notably very high for general Italian. As we can expect, the performances on setup1 were much lower than that for setup0 due to the intrinsic difficulties of parsing tweets and to the scarcity of annotated tweets for training. Joining the two datasets in the setup2 allowed to get a relevant gain in parsing tweets even if we added out-of-domain data. For these reasons, for all the following experiments, we abandoned the setup1 because it seemed more relevant to use the joined data (setup2) and compare them to setup0.

3 An Ensemble of Neural Parsers

The DEPENDABLE tool in Choi et al. (2015) reports ensemble upper bound performance assuming that, given the parsers outputs, the best tree can be identified by an oracle "MACRO" (MA), or that the best arc can be identified by another oracle "MICRO" (mi). Table 3 shows that, by applying these oracles, we have plenty of space for improving the performances by building some kind of ensemble system able to cleverly choose the correct information from the different parsers outputs and combine them improving the final solution. This observation motivates our proposal.

To combine the parser outputs we used the following ensemble schemas:

- **Voting**: Each parser contributes by assigning a vote on every dependency edge as described in Zeman and Žabokrtský (2005). With the majority approach the dependency tree could be ill-formed, in this case using the switching approach the tree is replaced with the output of the first parser.
- **Reparsing**: As described in Sagae and Lavie (2006) together with Hall et al. (2007) a MST algorithm is used to reparse a graph where

setup0				
	Valid. Ita		**Test Ita**	
	UAS	LAS	UAS	LAS
CM14	88.20/0.18	85.46/0.14	89.33/0.17	86.85/0.22
BA15	91.15/0.11	88.55/0.23	91.57/0.38	89.15/0.33
KG16:T	91.17/0.29	88.42/0.24	91.21/0.33	88.72/0.24
KG16:G	91.85/0.27	89.23/0.31	92.04/0.18	89.65/0.10
AN16	85.52/0.34	77.67/0.30	87.70/0.31	79.48/0.24
CH16	92.42/0.00	89.60/0.00	92.82/0.00	90.26/0.00
DM17	**93.37**/0.27	**91.37**/0.24	**93.72**/0.14	**91.84**/0.18
SH17	89.67/0.24	85.05/0.24	89.89/0.29	84.55/0.30
NG17	90.37/0.12	87.19/0.21	90.67/0.15	87.58/0.11
setup1				
	Valid. PoSTW		**Test PoSTW**	
	UAS	LAS	UAS	LAS
CM14	81.03/0.17	75.24/0.30	81.50/0.28	76.07/0.17
BA15	83.44/0.20	77.70/0.25	84.06/0.38	78.64/0.44
KG16:T	77.38/0.14	68.81/0.25	77.41/0.43	69.13/0.43
KG16:G	78.81/0.23	70.14/0.33	78.78/0.44	70.52/0.51
AN16	77.74/0.25	66.63/0.16	77.78/0.33	67.21/0.30
CH16	84.78/0.00	78.51/0.00	86.12/0.00	79.89/0.00
DM17	**85.01**/0.16	**78.80**/0.09	**86.26**/0.16	**80.40**/0.19
SH17	80.52/0.18	73.71/0.14	81.11/0.29	74.53/0.26
NG17	82.02/0.11	75.20/0.24	82.74/0.39	76.22/0.41
setup2				
	Valid. Ita+PoSTW		**Test PoSTW**	
	UAS	LAS	UAS	LAS
CM14	85.52/0.13	81.51/0.05	82.62/0.24	77.45/0.23
BA15	87.85/0.13	83.80/0.12	85.15/0.29	80.12/0.27
KG16:T	83.89/0.23	77.77/0.26	80.47/0.36	72.92/0.46
KG16:G	84.70/0.14	78.41/0.14	81.41/0.37	73.49/0.19
AN16	82.95/0.33	73.46/0.37	79.81/0.27	69.19/0.19
CH16	89.16/0.00	84.56/0.00	86.85/0.00	80.93/0.00
DM17	**89.72**/0.10	**85.85**/0.13	**87.22**/0.24	**81.65**/0.21
SH17	85.85/0.36	80.00/0.39	83.12/0.50	76.38/0.38
NG17	86.81/0.04	82.13/0.09	84.09/0.07	78.02/0.11

Table 2: Mean/standard deviation of UAS/LAS for each parser and for the different setups by repeating the experiments 5 times. All the results are statistically significant ($p < 0.05$) and the best values are showed in boldface.

	Validation		**Test**	
	UAS	LAS	UAS	LAS
setup0				
mi	98.30%	97.82%	98.08%	97.72%
MA	96.62%	95.10%	96.31%	94.82%
setup2				
mi	97.08%	96.02%	96.32%	94.73%
MA	94.62%	91.29%	93.27%	88.50%

Table 3: Results obtained by building an ensemble system based on the oracles mi e MA and considering all parsers.

each word in the sentence is a node. The MSTs algorithms used are Chu-Liu/Edmons (cle) and Eisner as reported in McDonald et al. (2005). Three weighting strategies for Chu-Liu/Edmons are used: equally weighted (w2); weighted according to the total labeled accuracy on the validation set (w3); weighted according to labeled accuracy per coarse grained PoS tag on the validation set (w4).

- **Distilling**: In Kuncoro et al. (2016) the authors train a distillation parser using a loss objective with a cost that incorporates ensemble uncertainty estimates for each possible attachment.

4 Results

Tables 4, 7 and 9 show the performances of the ensembles built on the best results on validation set obtained in the 5 training/test cycles considering both setup0 and setup2. Table 6 reports the number of malformed trees for the majority strategy.

Table 5 and 8 report the number of cases when the ensemble combination output differs from the baseline, including both labeled (L) and unlabeled (U) outputs. On the average the percentage of different unlabeled output varies from 2% to 15% with respect to baseline. For the best result (DM17+ALL) the difference on setup0 and setup2 is about 4%.

The results of the voting approach reported in Table 4 shows that the majority strategy is slightly better than the switching strategy, although it must be taken into account that there might be ill-formed dependency trees for the former strategy. The percentage of ill-formed trees on valid./test set vary from a minimum of 2% to a maximum of 8%. For this reasons the majority strategy should be used when it is followed by a manual correction phase. The switching strategy performs well if the first parser of voters is one of the best parsers, in fact the combinations AN16+ALL and AN16+CM14+SH17 have worst performance than the counterparts which using the best parser (DM17) as the first voter. Overall, the highest performance is achieved using all parsers together with DM17 as the first voter. For setup0 the increases are +0.19% in UAS e +0.38% in LAS, while in setup2 are +0.92% in UAS e +2.47% in LAS with respect to the best single parser (again DM17).

The results of the reparsing approach reported in Table 7 shows that the Chu-Liu/Edmonds algorithm is slightly better than the Eisner algorithm. In this case, the choice of which strategy

setup0				
	Validation		Test	
Voters/Strategy	UAS	LAS	UAS	LAS
DM17+CH16+BA15/maj.	94.20%	92.27%	93.77%	92.13%
DM17+CH16+BA15/swi.	94.11%	92.16%	93.79%	92.14%
AN16+CM14+SH17/maj.	90.43%	87.96%	91.03%	88.47%
AN16+CM14+SH17/swi.	89.44%	86.77%	90.17%	87.43%
DM17+CM14+SH17/maj.	93.84%	92.03%	93.82%	92.27%
DM17+CM14+SH17/swi.	93.76%	91.94%	93.82%	92.25%
AN16+ALL/maj.	94.37%	92.65%	93.83%	92.27%
AN16+ALL/swi.	93.99%	92.15%	93.43%	91.73%
DM17+ALL/maj.	**94.42**%	**92.67**%	**93.94**%	**92.41**%
DM17+ALL/swi.	94.38%	92.60%	93.91%	92.37%
DM17 (baseline)	93.74%	91.66%	93.75%	92.03%
setup2				
	Validation		Test	
Voters/Strategy	UAS	LAS	UAS	LAS
DM17+CH16+BA15/maj.	90.57%	87.16%	88.21%	83.64%
DM17+CH16+BA15/swi.	90.51%	87.10%	88.13%	83.51%
AN16+CM14+SH17/maj.	86.90%	83.60%	84.09%	79.78%
AN16+CM14+SH17/swi.	86.01%	82.50%	82.58%	77.94%
DM17+CM14+SH17/maj.	90.35%	87.21%	88.07%	83.64%
DM17+CM14+SH17/swi.	90.27%	87.11%	87.99%	83.52%
AN16+ALL/maj.	90.30%	87.26%	88.36%	84.13%
AN16+ALL/swi.	89.70%	86.45%	87.46%	83.06%
DM17+ALL/maj.	90.64%	87.60%	**88.51**%	**84.42**%
DM17+ALL/swi.	**90.65**%	**87.62**%	88.50%	84.20%
DM17 (baseline)	89.82%	85.96%	87.59%	81.95%

Table 4: Results of ensembles using switching and majority approaches on the best models in setup0 and setup2. The baseline is defined by the best results of Dozat and Manning (2017).

setup0				
	Validation		Test	
	/11.908		/10.417	
Voters/Strategy	U	L	U	L
DM17+CH16+BA15/maj.	208	61	188	46
DM17+CH16+BA15/swi.	192	52	175	39
AN16+CM14+SH17/maj.	1.006	424	783	336
AN16+CM14+SH17/swi.	1.130	489	870	371
DM17+CM14+SH17/maj.	170	37	139	15
DM17+CM14+SH17/swi.	157	33	129	13
AN16+ALL/maj.	382	126	328	105
AN16+ALL/swi.	460	164	386	133
DM17+ALL/maj.	356	117	282	81
DM17+ALL/swi.	312	97	255	72
setup2				
	Validation		Test	
	/24.243		/12.668	
Voters/Strategy	U	L	U	L
DM17+CH16+BA15/maj.	597	219	470	213
DM17+CH16+BA15/swi.	521	185	394	172
AN16+CM14+SH17/maj.	2.757	1.329	1.805	941
AN16+CM14+SH17/swi.	2.976	1.429	1.986	1.033
DM17+CM14+SH17/maj.	490	140	337	93
DM17+CM14+SH17/swi.	453	121	300	73
AN16+ALL/maj.	1.377	624	897	440
AN16+ALL/swi.	1.610	741	1.063	534
DM17+ALL/maj.	1.156	502	784	378
DM17+ALL/swi.	920	374	614	280

Table 5: Numbers of cases when there is a different output between the ensemble systems, using switching and majority, and the baseline Dozat and Manning (2017).

to use must take into account if we want to allow non-projectivity or not. The percentage of non-projective dependency trees on valid./test set for Chu-Liu/Edmonds vary from a minimum of 7% to a maximum of 12% compared with the average for the Italian corpora of 4%. Overall, the highest performances are achieved using Chu-Liu/Edmonds algorithm. For setup0 the increases are +0.25% in UAS and +0.45% in LAS, while in setup2 are +0.77% in UAS and +2.30% in LAS with respect to the best single parser (DM17).

The results of the distilling strategy reported in Table 9, unlike the previous proposals, show worse outcomes, which score below the baseline.

5 Discussion and Conclusions

We have studied the performances of some neural dependency parsers on generic and social media domain. Using the predictions of each single parser we combined the best outcomes to improve the performance in various ways. The ensemble models are more efficient on corpora built using in-domain data (social media), giving an improvement of $\sim 1\%$ in UAS and $\sim 2.5\%$ in LAS.

	setup0		setup2	
Voters	**Valid.** /564	**Test** /482	**Valid.** /1235	**Test** /674
DM17+CH16+BA15	9	7	31	31
AN16+CM14+SH17	45	25	88	77
DM17+CM14+SH17	6	6	19	23
AN16+ALL	18	17	73	63
DM17+ALL	17	11	75	57

Table 6: Number of malformed trees obtained by using the majority strategy for both setups.

Thanks to the number of parser models adopted in the experiments it has been possible to verify that the performances of the ensemble models increase as the number of parsers grows.

The improvement of LAS is, in most cases, at least twice the value of UAS. This could mean that ensemble models catch with better precision the type of dependency relations rather than head-dependent relations.

All the proposed ensemble strategies, except for distilling, perform more or less in the same way, therefore the choice of which strategy to use is due, in part, to the properties that we want to obtain on the combined dependency tree.

Our work is inspired by the work of Mazzei

setup0				
	Validation		Test	
Voters/Strategy	UAS	LAS	UAS	LAS
DM17+CH16+BA15/cle-w2	93.82%	91.85%	93.54%	91.83%
DM17+CH16+BA15/cle-w3	93.89%	91.82%	93.78%	92.06%
DM17+CH16+BA15/cle-w4	94.20%	92.28%	93.72%	92.04%
DM17+CH16+BA15/eisner	94.05%	92.05%	93.46%	91.78%
ALL/cle-w2	**94.31**%	92.53%	93.85%	92.23%
ALL/cle-w3	94.16%	92.41%	**94.00**%	**92.48**%
ALL/cle-w4	94.29%	**92.58**%	93.95%	92.38%
ALL/eisner	**94.31**%	92.53%	93.95%	92.35%
DM17 (baseline)	93.74%	91.66%	93.75%	92.03%
setup2				
	Validation		Test	
Voters/Strategy	UAS	LAS	UAS	LAS
DM17+CH16+BA15/cle-w2	90.33%	86.95%	87.69%	83.31%
DM17+CH16+BA15/cle-w3	89.82%	85.96%	87.59%	81.95%
DM17+CH16+BA15/cle-w4	90.41%	86.99%	87.94%	83.32%
DM17+CH16+BA15/eisner	90.50%	87.05%	88.04%	83.51%
ALL/cle-w2	**90.52**%	**87.53**%	**88.36**%	**84.25**%
ALL/cle-w3	89.90%	86.75%	87.79%	83.54%
ALL/cle-w4	90.42%	87.46%	88.19%	84.11%
ALL/eisner	90.45%	87.41%	88.31%	84.08%
DM17 (baseline)	89.82%	85.96%	87.59%	81.95%

Table 7: Results of ensembles using reparsing approaches on the best models in setup0 and setup2. The baseline is again defined by the best results of DM17.

setup0				
	Validation		Test	
	/11.908		/10.417	
Voters/Strategy	UAS	LAS	UAS	LAS
DM17+CH16+BA15/cle-w2	360	129	307	90
DM17+CH16+BA15/cle-w3	96	0	89	1
DM17+CH16+BA15/cle-w4	267	76	247	52
DM17+CH16+BA15/eisner	375	130	327	103
ALL/cle-w2	400	131	333	103
ALL/cle-w3	351	108	299	79
ALL/cle-w4	383	126	307	87
ALL/eisner	411	133	333	106
setup2				
	Validation		Test	
	/24.243		/12.668	
Voters/Strategy	UAS	LAS	UAS	LAS
DM17+CH16+BA15/cle-w2	1.056	496	800	424
DM17+CH16+BA15/cle-w3	0	0	0	0
DM17+CH16+BA15/cle-w4	603	264	491	236
DM17+CH16+BA15/eisner	1.047	443	789	376
ALL/cle-w2	1.347	599	882	417
ALL/cle-w3	1.261	537	804	363
ALL/cle-w4	1.274	576	822	389
ALL/eisner	1.367	607	916	436

Table 8: Numbers of cases when there is a different output between the ensemble systems, using reparsing approaches, and the baseline Dozat and Manning (2017).

(2015). Different from his work, we use larger set of state-of-the-art parsers, all based on neural networks, in order to gain more diversity among the models used in the ensembles; furthermore we have experimented the distilling strategy and eisner reparsing algorithm. Moreover, we built ensembles on larger datasets using both generic and social media texts.

Setup	UAS	LAS
setup0	92.50% (–1.25%)	89.93% (–2.10%)
setup2	86.73% (–0.86%)	81.39% (–0.56%)

Table 9: Results of distilling approach on the best models in setup0 and setup2. In brackets are reported the differences between the distilled models and the best results of DM17, as baseline.

Acknowledgements

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

References

Anita Alicante, Cristina Bosco, Anna Corazza, and Alberto Lavelli. 2015. Evaluating italian parsing across syntactic formalisms and annotation schemes. In Roberto Basili, Cristina Bosco, Rodolfo Delmonte, Alessandro Moschitti, and Maria Simi, editors, *Harmonization and Development of Resources and Tools for Italian Natural Language Processing within the PARLI Project*, Springer International Publishing, Cham, pages 135–159.

Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov, and Michael Collins. 2016. Globally normalized transition-based neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. ACL, Berlin, Germany, pages 2442–2452.

Miguel Ballesteros, Chris Dyer, and Noah A. Smith. 2015. Improved transition-based parsing by modeling characters instead of words with lstms. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. ACL, Lisbon, Portugal, pages 349–359.

Cristina Bosco, Felice DellOrletta, Simonetta Montemagni, Manuela Sanguinetti, and Maria Simi. 2014. The evalita 2014 dependency parsing task. In *Proceedings of the Fourth Inter-*

national Workshop EVALITA 2014. Pisa, Italy, pages 1–8.

Cristina Bosco and Alessandro Mazzei. 2011. The evalita 2011 parsing task. In *Working Notes of EVALITA 2011*, CELCT, Povo, Trento.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. ACL, Doha, Qatar, pages 740–750.

Hao Cheng, Hao Fang, Xiaodong He, Jianfeng Gao, and Li Deng. 2016. Bi-directional attention with agreement for dependency parsing. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. ACL, Austin, Texas, pages 2204–2214.

Jinho D. Choi, Joel Tetreault, and Amanda Stent. 2015. It depends: Dependency parser comparison using a web-based evaluation tool. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. ACL, Beijing, China, pages 387–396.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of the 2017 International Conference on Learning Representations*.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. 2015. Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. ACL, Beijing, China, pages 334–343.

Johan Hall, Jens Nilsson, Joakim Nivre, Gülsen Eryigit, Beáta Megyesi, Mattias Nilsson, and Markus Saers. 2007. Single malt or blended? a study in multilingual parser optimization. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*. ACL, Prague, Czech Republic, pages 933–939.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional lstm feature representations. *Transactions of the Association for Computational Linguistics* 4:313–327.

Adhiguna Kuncoro, Miguel Ballesteros, Lingpeng Kong, Chris Dyer, and Noah A. Smith. 2016. Distilling an ensemble of greedy dependency parsers into one mst parser. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. ACL, Austin, Texas, pages 1744–1753.

Alberto Lavelli. 2013. An ensemble model for the evalita 2011 dependency parsing task. In Bernardo Magnini, Francesco Cutugno, Mauro Falcone, and Emanuele Pianta, editors, *Evaluation of Natural Language and Speech Tools for Italian*. Springer Berlin Heidelberg, Berlin, Heidelberg, pages 30–36.

Alberto Lavelli. 2014. Comparing state-of-the-art dependency parsers for the evalita 2014 dependency parsing task. In *Proceedings of the Fourth International Workshop EVALITA 2014*. Pisa, Italy, pages 15–20.

Alberto Lavelli. 2016. Comparing state-of-the-art dependency parsers on the italian stanford dependency treebank. In *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC-it 2016)*. Napoli, Italy, pages 173–178.

Alessandro Mazzei. 2015. Simple voting algorithms for italian parsing. In Roberto Basili, Cristina Bosco, Rodolfo Delmonte, Alessandro Moschitti, and Maria Simi, editors, *Harmonization and Development of Resources and Tools for Italian Natural Language Processing within the PARLI Project*, Springer International Publishing, Cham, pages 161–171.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajic. 2005. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*. ACL, Vancouver, British Columbia, Canada, pages 523–530.

Dat Quoc Nguyen, Mark Dras, and Mark Johnson. 2017. A novel neural network model for joint pos tagging and graph-based dependency parsing. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. ACL, Vancouver, Canada, pages 134–142.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. ACL, Copenhagen, Denmark, pages 338–348.

Kenji Sagae and Alon Lavie. 2006. Parser combination by reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*. ACL, Stroudsburg, PA, USA, NAACL-Short '06, pages 129–132.

Tianze Shi, Liang Huang, and Lillian Lee. 2017a. Fast(er) exact decoding and global training for transition-based dependency parsing via a minimal feature set. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. ACL, Copenhagen, Denmark, pages 12–23.

Tianze Shi, Felix G. Wu, Xilun Chen, and Yao Cheng. 2017b. Combining global models for parsing universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. ACL, Vancouver, Canada, pages 31–39.

Daniel Zeman and Zdeněk Žabokrtský. 2005. Improving parsing accuracy by combining diverse dependency parsers. In *Proceedings of the Ninth International Workshop on Parsing Technology*. ACL, Vancouver, British Columbia, pages 171–178.

"*Buon appetito!*" - Analyzing Happiness in Italian Tweets

Pierpaolo Basile and **Nicole Novielli**
Department of Computer Science
University of Bari Aldo Moro
Via, E. Orabona, 4 - 70125 Bari (Italy)
{firstname.lastname}@uniba.it

Abstract

English. We report the results of an exploratory study aimed at investigating the language of happiness in Italian tweets. Specifically, we conduct a time-wise analysis of the happiness load of tweets by leveraging a lexicon of happiness extracted from 8.6M tweets. Furthermore, we report the results of a statistical linguistic analysis aimed at extracting the most frequent concepts associated with the happy and sad words in our lexicon.

Italiano. *Riportiamo i risultati dell'analisi esplorativa di un corpus di tweet in Italiano, al fine di individuare i concetti tipicamente associati alla felicità. Riportiamo inoltre i risultati di un'analisi time-wise dell'happiness load dei tweet nelle diverse ore della giornata e nei diversi giorni della settimana.*

1 Introduction

The widespread diffusion of social media has reshaped the way we interact and communicate. Among others, microblogging platforms as Twitter are becoming extremely popular and people constantly use them for sharing opinions about facts of public interest. Furthermore, its worldwide adoption and the fact that tweets are publicly available, makes Twitter an extremely appealing virtual place for researchers interested in language analysis as a mean to investigate social phenomena (Bollen et al., 2009; Garimella et al., 2016).

In addition, recent research showed how microblogging is also used for self-disclosure of individual feelings (Roberts et al., 2012; Andalibi et al., 2017). As such, microblogs constitute an invaluable wealth of data ready to be mined for discovering affective stereotypes (Joseph et al., 2017) using corpus-based approaches to linguistic ethnography (Mihalcea and Liu, 2006). Such analyses, can further enhance our understanding on how people conceptualize the experience of emotions and what are their more common triggers. Recent studies even envisaged the emergence of tools for monitoring the public mood [1] and health through the analysis of Twitter users' reaction to major social, political, economics events (Bollen et al., 2009).

In this study we report the results of an exploratory analysis of the language of happiness in Twitter. In particular, we perform a partial replication of the approach proposed by (Mihalcea and Liu, 2006) for mining sources of happiness in blog posts. The contributions of this paper are as follows. First, we extract a happiness dictionary from a sample of about 8.6M tweets from the TWITA corpus of Italian tweets (Basile and Nissim, 2013). For each word in the dictionary, we compute a *happiness factor* by adapting the approach proposed in the original study. Furthermore, we perform a qualitative investigation of the 100 happiest and saddest words by mapping them into psycholinguistic word categories (see Section 2). As a second step, we use our dictionary to perform a time-wise analysis of happiness as shared in different hours and days of the week (see Section 3). Third, we extract concepts most frequently associated with happy words in our dictionary, which we map into WordNet super-senses (see Section 4). We discuss limitations and provide suggestions for future work in Section 5.

2 The Happiness Dictionary

2.1 A Dataset of Happy and Sad Tweets

Our study is based on TWITA (Basile and Nissim, 2013), the largest available corpus of Ital-

[1] 'What Twitter tells us about our happiness' https://goo.gl/fmYBP3 - Last accessed: Oct. 2018

ian tweets. In particular, we analyze a subset of 400M tweets obtained by filtering-out re-tweets from all the 500M tweets collected from February 2012 to September 2015. Following the idea proposed in (Read, 2005; Go et al., 2009), we select positive and negative tweets based on the presence of positive or negative emoticons[2]. Since a tweet can contain multiple emoticons, we selected only tweets that contain a single emoticon appearing at the end of the tweet. Using this procedure we obtain a corpus C_{happy} of 8,648,476 tweets.

2.2 Happy/Sad Word Extraction and Scoring

From the C_{happy} corpus, we extract a subset of words and we assign them an happiness factor (hf) computed according to the log of the odds ratio between the probability that the word occurs in positive tweets $p_{happy}(w_i)$ and the probability that it occurs in negative tweets $p_{sad}(w_i)$ as in Eq. 1.

$$hf(w_i) = log\frac{p_{happy}(w_i)}{p_{sad}(w_i)} \quad (1)$$

We adopt additive smoothing (Laplace smoothing) for computing both p_{happy} and p_{sad} probabilities. In our lexicon, we include and compute the happiness factor only for words that occur at least 10,000 times, for a total of 718 words. We call this list "the happiness dictionary" (D_h)[3]. Table 1 reports the most happy/sad words with the corresponding happiness factor (score(hf)).

Table 1: The happiness factor of the most happy/sad words.

happy	**score (hf)**	**sad**	**score (hf)**
fback	4.04	triste	-2.37
ricambi	3.83	purtroppo	-1.91
benvenuta	3.17	dispiace	-1.68
grazie	2.32	brutto	-1.68
buon	2.14	peccato	-1.63
piacere	2.03	manca	-1.53
gentile	1.91	compiti	-1.35
auguro	1.86	paura	-1.33
dolcezza	1.74	studiare	-1.30

We observe that some happy words *(fback, ricambi, benvenuta)* are due to several positive tweets that users post when they establish new connections, i.e. when they start following a new user or when they ask sombebody to follow them back *(fback)* as in: *@usermention ciao sono nuova, fback? Grazie mille :)* Sad words refer to negative emotions or evaluations, such as *triste*, *dispiace*, *brutto*, *peccato*. Interestingly, several negative words emerge from the school domain (*compiti*, *studiare*) and the word *scuola* has a negative score of -0.93 itself.

2.3 Happiness by Psycholinguistic Categories

We are interested in understanding how happiness words map into psycholinguistic word classes. Hence, we check their distribution along the word categories in the Linguistic Inquiry and Word Count (LIWC) taxonomy (Pennebaker and Francis, 2001). To this aim, we perform a qualitative investigation on the 100 most happy and 100 most sad words, that are the words with the highest and lowest happiness scores, respectively. We map each word into LIWC word categories. LIWC organizes words into psychologically meaningful categories, based on the assumption that the language reflects the cognitive and emotional phenomena involved in communication. It has been used for a wide range of psycholinguistics experimental settings, including investigation on emotions, social relationships, and thinking styles (Tausczik and Pennebaker, 2010).

We perform a coding of the English translation of the happy/sad words into LIWC categories. When translating, we keep the information about the subject conveyed by the Italian verbs (e.g., 'penso' is translated as 'I think'). The coding is performed manually by the authors: in a first round, one rater associates each word with the corresponding LIWC category; then, the other revises the annotation, checking for consistency and verifying also the correctness of the translation. 22 words are discarded and replaced with others from the dictionary because we could not find a mapping with any of the categories. Furthermore, we add an *ad hoc* category to enable modeling of words from the social media domain (*retweet*, *follow*).

Figure 1 shows how the happy and sad words distribute along the dimensions associated with the most frequent categories. Sample words for each word category are reported in Table 2. We observe that happy words in the dictionary mainly refer to positive emotions as well as to the social and social media dimensions. Conversely, sad words mainly

[2]We use :-) and :) for happy and :-(and :(for sad.

[3]The dictionary is available on github `https://github.com/pippokill/happyFactor`

describe negative emotions with focus on the author. Words describing cognitive mechanisms are also associated with sadness.

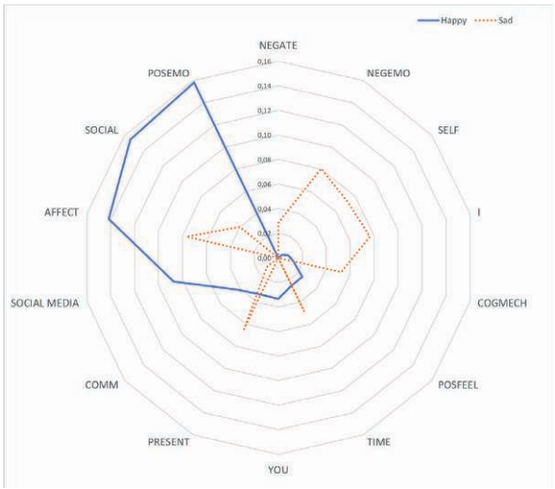


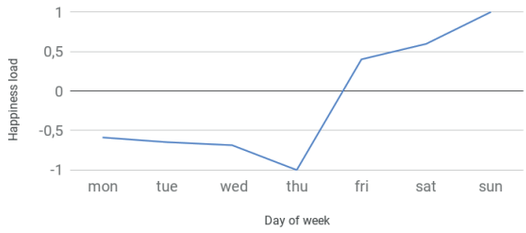
Figure 1: Comparing the most happy/sad words along dimensions associated with word categories.

Table 2: Mapping the happiness dictionary to word categories

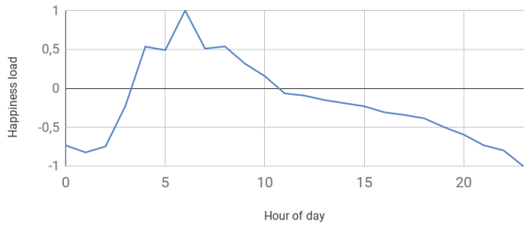
Category	Sample words
Affect	buono/a, ottimo, triste, brutto
Cogmech	avrei, pensare, capisco, so, volevo
Comm	benvenut*, buonanotte, ciao
I	mi, io, *first person verbs*
Negate	mai, nulla, non
Negemo	difficile, peggio, sola
Posemo	benvenuta, piacere, sorriso, cara
Posfeel	cara, contenta, adoro, felice
Present	avermi, trovi, riesco
Self	mi, io, *first person verbs*
Social	ricambi, gruppo
S. media	fback, follow, seguire, Instagram
Time	serata, anticipo, periodo, ultima
You	te, tuo, *second person verbs*

3 Time-wise analysis

As observed in the original study, happiness is not constant in our life and different degrees of happiness might be observed at different moments in time. As such, we analyze how happiness changes over time. In particular we take into account the days of the week and the different hours in a day. For this analysis, we exploit the whole corpus of 400M tweets and we compute the distribution



(a) Happiness load by day of the week



(b) Happiness load for a 24-hour day

Figure 2: Time-wise analysis.

of words occurring in the happiness dictionary in each different time period. Using this strategy, in each time period the word has an happiness load obtained by multiplying its frequency in that period by its happiness factor. The happiness load of each time period is the average of all the happiness load in that period. The obtained values are mapped in the interval [-1, 1] and plotted in Figure 2a (for days) and in Figure 2b (for hours).

Our time-wise analysis reveals a drop in happiness on Thurdsay, with a subsequent twist towards positive mood on Friday, before the weekend that is the happiest moment in the week. This is consistent with the findings of the original study reporting mid-week blues around Wednesday and a happiness peak on Saturday (Mihalcea and Liu, 2006). Regarding the hours, we observe the highest happiness load in the morning, with a peak around 6 AM, and it constantly decreases over the day, with the lowest value observed around 11 PM.

4 Concept analysis

We are interested in concepts related to words in the happiness dictionary. In the original study, the authors extract the 'ingredients' for their recipe of happiness by ranking the most relevant 2- and 3-grams from their corpus according to their happiness load. Such an approach is not easy to replicate as the number of 2- and 3-grams extracted from 400M tweets is potentially huge. Hence, starting from the words in our happiness dictio-

Table 3: The most happy and sad word pairs.

	word pair	score
happy	buon, appetito	9.74
	buon, auspicio	8.84
	dolcezza, infinita	6.94
	grazie, mille	5.23
	piacere, ciao	5.12
	grazie, esistere	4.50
sad	dispiacere, deludervi	-9.28
	brutto, presentimento	-8.45
	triste, arrabbiata	-8.10
	peccato, potevamo	-4.85
	triste, piangere	-3.68
	studiare, matematica	-3.55
	peccato, gola	-2.63
	manca, vederlo	-1.97

nary, we extract the most 50 co-occurring words in a window of two words. Then we rank all the word pairs (dictionary word, co-occurring word)[4] according to the Pointwise Mutual Information (PMI) multiplied by the happiness factor. Table 3 reports some of the most happy and sad pairs.

Starting from word pairs, we perform another kind of analysis aiming at mapping the words occurring in each pair with super-senses in WordNet. A super-sense is a general semantic taxonomy defined by the WordNet lexicographer classes as a way for defining logical aggregation of senses in each syntactic category. We assign a happiness score to each super-sense by averaging the happiness factor associated with the dictionary word in the pair. Since each pair contains a dictionary word and a co-occurring word, we map the co-occurring word to its super-sense and increment the score of the super-sense by summing the happiness factor associated with the dictionary word. Finally, the score of each super-sense is divided by the number of the co-occurring words belonging to the super-sense. For ambiguous words, we select the super-sense associated with the most frequent sense. In this study, we do not rely on a Word Sense Disambiguation (WSD) algorithm since WSD is a critical task. We need to test the WSD performance on tweets before to use it. Generally, WSD algorithms give performance slightly above the most frequent sense. We plan to test WSD in a further study. As super-senses are defined in the English version of WordNet, we performed a mapping of Italian words to the English WordNet through the use of both Morph-it! (Zanchetta and Baroni, 2005) and MultiWordNet (Pianta et al., 2002), while sense occurrences are extracted from MultiSemCor (Bentivogli and Pianta, 2005).

In Table 4 we report the most happy and sad super-senses with the most frequent words extracted by our corpus. Consistently with the evidence provided by the analysis of the psycholinguistic word categories (see Section 2.3), we observe that socialness is associated with positive feelings, with concepts referring to people (*noun.person*) and communication (*verb.communication*, *noun.communication*) scoring high in happiness. Food (*noun.food*) also seems to be a major cause of positive mood, as well as money and gifts (*noun.possession*), sport achievements (*'vittoria* and *'gol'* in *noun.act*), and mundane locations and events (*'centro', 'piazza', 'concerto', 'viaggio'* in *noun.location* and *noun.act*). This is consistent with suggestion by (Mihalcea and Liu, 2006) to enjoy food and drinks in an 'interesting social place' as a recipe for happiness. People also report their desires and preferences (*voglio, amo, spero* in *verb.emotion*).

Also for sadness, results confirm findings emerging from the analysis of psycholinguistic categories in LIWC. In fact, we observe that people tend to report their own individual negative feelings (*rido, piango* in *verb.body*), thoughts (*verb.cognition*), perceptions (e.g., *'vedo', 'sento'*), and personal needs (*'bisogno'* and *'sonno'* in *noun.state*). We observe also stereotypical complaints about weather (*piove*) as well as swear words (*noun.body*).

5 Discussion and Conclusions

We performed an exploratory analysis of the lexicon and concepts associated with happiness in Italian tweets. We leveraged a corpus of happy and sad tweets to extract a "happiness dictionary', which we use to perform a time-wise analysis of happiness on Twitter and to extract the most frequent concepts and psycholinguistic categories associated to positive and negative emotions.

This study is a partial replication of the previous one by (Mihalcea and Liu, 2006) on blog posts. The main differences with respect to the original study are in the size, language and source of the corpus used for extracting the happiness

[4]We do not take into account the word order in the pairs.

Table 4: The most happy and sad super-senses based in our corpus.

	super-sense	**most frequent concepts**
happy	noun.relation	resto, ricambio
	noun.food	cena, pranzo, colazione, caffé
	noun.attribute	coraggio, voce, numero, bellezza, splendore, silenzio
	noun.person	mamma, ragazz*, amic*, dio, tesoro, donna
	verb.communication	dico(no), parlare, prego, profilo, parla, chiedere
	noun.communication	film, scusa, merda, musica, buongiorno, canzone, concerto
	verb.possession	trov*, dare, perdere, perso, averti, comprato
	verb.emotion	voglio/vorrei, amo, piace, vuoi, spero, odio, auguri
	noun.location	sito, centro, post, piazza, scena, sud, nord, regione
	noun.possession	soldi, regalo, fondo
	noun.event	vittoria, gara, onda, campagna, scarica, fuoco, episodio, meraviglia
	noun.act	cose, partita, gol, colpa, ricerca, viaggio, tour, bacio, corso, sesso
sad	verb.consumption	bisogna, mangiare, usare, mangio/mangiato, usa/o, usato, mangio
	verb.body	piangere, dormire, ridere, sveglia, sorridere, piango, rido
	noun.body	*swear words*, testa, occhi, mano/i, capelli
	verb.change	inizio/inizia(re), cambiare, finito, morire/morte, successo, finisce
	verb.perception	vedere, vedo, sento, sentire, guarda, guardare, ascoltare, pare
	verb.cognition	so, sai, penso, letto, credo, sa, leggere, sapere, pensare, studiare
	noun.state	bisogno, punto, problemi/a, accordo, pace, crisi, situazione, sonno
	noun.substance	aria, acqua
	verb.weather	piove

lexicon. Specifically, (Mihalcea and Liu, 2006) rely on a collection of 10,000 blog posts in English from LiveJournal.com to extract a list of happy/sad words with their associated happiness scores, while we leverage a bigger corpus consisting of 8.6M Italian tweets. Furthermore, the blog posts were labeled as happy or sad by their authors. Conversely, for tweets we relied on silver labeling based on the presence of emoticons as a proxy the author self-reporting of her own positive or negative emotions.

Our analysis of psycholinguistic categories and the extraction of concepts and WordNet super-senses associated with them reveals interesting findings. Happiness appears related to the social aspects of life while sad tweets mainly revolves around self-centered negative feelings and thoughts. In addition, our-time wise analysis reveals a mid-week drop in happiness also observed in the original study. We also observe that happiness is high in the morning and decreases over the day. As a future work, it would be interesting to investigate if time-wise analysis based on hours produces consistent results if a weekday or the weekend is considered and if emotion-triggering concepts associated with happiness also vary over time.

We are aware of the main limitations of this study. First of all, by relying on microblogs we are probably able to mine emotion triggers that do not necessarily coincide with those shared in daily face-to-face conversations or reported in private logs. Furthermore, we do not attempt to make any categorization of the authors of tweets. Indeed, different target user groups could be studied to fulfill specific research goals and enable perspective applications, i.e. for supporting creative writing or for providing personalized recommendations based on moods. Finally, we consider only Twitter as a source of data. The same methodology could produce different results if applied to other social media. Indeed, recent research (Andalibi et al., 2017) showed that other media, such as Instagram, are also used for sharing extremely private emotions, such as feelings linked to depression. Based on these observations, further replications could focus on finer-grained emotions, also leveraging corpora from different platforms and including consideration of demographics and geographical information (Mitchell et al., 2013; Allisio et al., 2013) as additional dimensions of analysis.

References

[Allisio et al.2013] Leonardo Allisio, Valeria Mussa, Cristina Bosco, Viviana Patti, and Giancarlo Ruffo. 2013. Felicittà: Visualizing and estimating happiness in italian cities from geotagged tweets. In *Proceedings of the First International Workshop on Emotion and Sentiment in Social and Expressive Media: approaches and perspectives from AI (ESSEM 2013) A workshop of the XIII International Conference of the Italian Association for Artificial Intelligence (AI*IA 2013), Turin, Italy, December 3, 2013.*, pages 95–106.

[Andalibi et al.2017] Nazanin Andalibi, Pinar Ozturk, and Andrea Forte. 2017. Sensitive self-disclosures, responses, and social support on instagram: The case of #depression. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, CSCW '17, pages 1485–1500, New York, NY, USA. ACM.

[Basile and Nissim2013] Valerio Basile and Malvina Nissim. 2013. Sentiment analysis on Italian tweets. In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 100–107.

[Bentivogli and Pianta2005] Luisa Bentivogli and Emanuele Pianta. 2005. Exploiting parallel texts in the creation of multilingual semantically annotated resources: the multisemcor corpus. *Natural Language Engineering*, 11(3):247–261.

[Bollen et al.2009] Johan Bollen, Alberto Pepe, and Huina Mao. 2009. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. *CoRR*, abs/0911.1583.

[Garimella et al.2016] Kiran Garimella, Michael Mathioudakis, Gianmarco De Francisci Morales, and Aristides Gionis. 2016. Exploring controversy in twitter. In *Proceedings of the 19th ACM Conference on Computer Supported Cooperative Work and Social Computing Companion*, CSCW '16 Companion, pages 33–36, New York, NY, USA. ACM.

[Go et al.2009] Alec Go, Lei Huang, and Richa Bhayani. 2009. Twitter sentiment analysis. *Entropy*, 17:252.

[Joseph et al.2017] Kenneth Joseph, Wei Wei, and Kathleen M. Carley. 2017. Girls rule, boys drool: Extracting semantic and affective stereotypes from twitter. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing, CSCW 2017, Portland, OR, USA, February 25 - March 1, 2017*, pages 1362–1374.

[Mihalcea and Liu2006] Rada Mihalcea and Hugo Liu. 2006. A corpus-based approach to finding happiness. In *Proc. AAAI Spring Symposium and Computational Approaches to Weblogs*, page 6 pages.

[Mitchell et al.2013] Lewis Mitchell, Morgan R. Frank, Kameron Decker Harris, Peter Sheridan Dodds, and Christopher M. Danforth. 2013. The geography of happiness: Connecting twitter sentiment and expression, demographics, and objective characteristics of place. *PLOS ONE*, 8(5):1–15, 05.

[Pennebaker and Francis2001] J. Pennebaker and M. Francis. 2001. Linguistic inquiry and word count: Liwc. *Mahway: Lawrence Erlbaum Associates*, 71.

[Pianta et al.2002] Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. 2002. Multiwordnet: developing an aligned multilingual database. 1st gwc. In *Proceedings of the First International Conference on Global WordNet*.

[Read2005] Jonathon Read. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL student research workshop*, pages 43–48. Association for Computational Linguistics.

[Roberts et al.2012] Kirk Roberts, Michael A. Roach, Joseph Johnson, Josh Guthrie, and Sanda M. Harabagiu. 2012. EmpaTweet: Annotating and Detecting Emotions on Twitter. In Nicoletta C. Chair, Khalid Choukri, Thierry Declerck, Mehmet U. Dou gan, 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).

[Tausczik and Pennebaker2010] Yla R. Tausczik and James W. Pennebaker. 2010. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54.

[Zanchetta and Baroni2005] Eros Zanchetta and Marco Baroni. 2005. Morph-it!: a free corpus-based morphological resource for the italian language.

Long-term Social Media Data Collection at the University of Turin

Valerio Basile
University of Turin
basile@di.unito.it

Mirko Lai
University of Turin
mirko.lai@unito.it

Manuela Sanguinetti
University of Turin
msanguin@di.unito.it

Abstract

We report on the collection of social media messages — from Twitter in particular — in the Italian language that is continuously going on since 2012 at the University of Turin. A number of smaller datasets have been extracted from the main collection and enriched with different kinds of annotations for linguistic purposes. Moreover, a few extra datasets have been collected independently and are now in the process of being merged with the main collection. We aim at making the resource available to the community to the best of our possibility, in accordance with the Terms of Service provided by the platforms where data have been gathered from.

(Italian) In questo articolo descriviamo il lavoro di raccolta di messaggi — da Twitter in particolar modo — in lingua italiana che va avanti in maniera continuativa dal 2012 presso l'Università di Torino. Diversi dataset sono stati estratti dalla raccolta principale ed arricchiti con differenti tipi di annotazione per scopi linguistici. Inoltre, dataset ulteriori sono stati raccolti indipendentemente, e fanno ora parte della raccolta principale. Il nostro scopo è rendere questa risorsa disponibile alla comunità in maniera più completa possibile, considerati i termini d'uso imposti dalle piattaforme da cui i dati sono stati estratti.

1 Introduction

The online micro-blogging platform *Twitter*[1] has been a popular source for natural language data since the second half of the 2010's, due to the enormous quantity of public messages exchanged by its users, and the relative ease of collecting them through the official API.

Many researchers implemented systems to collect large datasets of tweets, and share them with the community. Among them, the Content-centered Computing group at the University of Turin[2] is maintaining a large, diversified collection of datasets of tweets in the Italian language[3]. However, although the Twitter datasets in Italian make the majority of our collection, over the years, and also in the recent past, several resources have been created in other languages and including data retrieved from other sources than Twitter.

In this paper, we report on the current status of the collection (Section 2) and we give an overview of several annotated datasets included in it (Section 3). Finally, we describe our current and future plans to make the data and annotations available to the research community (Section 4).

2 TWITA: Long-term Collection of Italian Tweets

The current effort to collect tweets in the Italian language started in 2012 at the University of Groningen (Basile and Nissim, 2013). Taking inspiration from the large collection of Dutch tweets by Tjong Kim Sang and van den Bosch (2013), Basile and Nissim (2013) implemented a pipeline to collect and automatically annotate a large set of tweets in Italian by leveraging the Twitter API. The process interrogates the *stream* API with a set of keywords designed to capture the Italian language and at the same time excluding other languages. At the time of its publishing, the resource contained about 100 million tweets in Italian in the first year (from February 2012 to February

[1] https://twitter.com/

[2] http://beta.di.unito.it/index.php/english/research/groups/content-centered-computing/people

[3] Some of the datasets included in this report and their methodology of annotation are described in Sanguinetti et al. (2014)

2013). The automatic collection, however, continued, and in 2015 was transferred from the University of Groningen to the University of Turin. From June 2018, a new filter based on the five Italian vowels has been added to the pipeline, along with the language filter provided by the Twitter API, which was not previously available, in order to limit the number of accidentally captured tweets in other languages. In the latest version of the data collection pipeline, a Python script employing the tweepy library[4] gathers JSON tweets using the following filter: track=["a","e","i","o","u"] and languages=["it"]. We stored the raw, complete JSON tweet structures in zipped files for backup. Meanwhile, we store the text and the most useful metadata (username , timestamp, geolocalization, retweet and reply status) in a relational database in order to perform efficient queries.

At the time of this writing, the collection comprises more than 500 million tweets in the Italian language, spanning 7 years (57 months) from February 2012 to July 2018. There are a few holes in the collection, sometimes spanning entire months, due to incidents involving the server infrastructure or changes in the Twitter API which required manual adjustment of the collection software. Figure 1 shows the percentage of days in each month for which the collection has data, at the time of this writing.

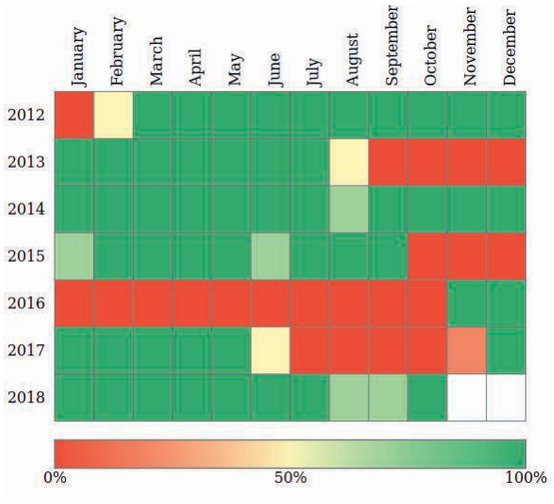


Figure 1: Percentage of days in each month for which tweets are available.

[4]http://www.tweepy.org/

3 Annotated Datasets

In the past years, the TWITA collection has been made available to many research teams interested in the study of social media in the Italian language with computational methods. Several such studies focused on creating new linguistic resources starting from the raw tweets and basic metadata provided by TWITA, including a number of datasets created for shared tasks of computational linguistics. In this section, we give an overview of such resources. Moreover, some datasets were created independently from TWITA, and are now managed under the same infrastructure, therefore we include them in this report.

For each dataset, we provide a summary infobox with basic information, including the type of annotation performed on the the dataset and how it was achieved, i.e., by means of expert annotators or a crowdsourcing platform.

3.1 Datasets From TWITA

The datasets described in this section are subsets of the main TWITA dataset, obtained by sampling the collection according to different criteria, and annotated for several purposes.

TWitterBuonaScuola (Stranisci et al., 2016) is a corpus of Italian tweets on the topic of the national educational and training systems. The tweets were extracted from a specific hashtag (#labuonascuola, the nickname of an education reform, translating to *the good school*) and a set of related keywords: "la buona scuola" (*the good school*), "buona scuola" (*good school*), "riforma scuola" (*school reform*), "riforma istruzione" (*education reform*).

Name: TWitterBuonaScuola
Size: 35,148 total tweets, 7,049 annotated tweets
Time period: February 22, 2014–December 31, 2014
Annotation: polarity, irony and topic
Annotation method: crowdsourcing
URL: http://twita.dipinfo.di.unito.it/tw-bs

TW-SWELLFER (Sulis et al., 2016) is a corpus of Italian tweets on subjective well-being, in particular regarding the topics of fertility and parenthood. The tweets were collected by searching for 11 hashtags — #papa (*father*), #mamma (*mother*), #babbo (*dad*), #incinta (*pregnant*), #primofiglio (*first child*), #secondofiglio (*second child*), #futuremamme (*future moms*), #maternita (*materhood*), #paternità (*fatherhood*), #allattamento (*nursing*), #gravi-

danza (*pregnancy*) — and 19 related keywords.

Name: TW-SWELLFER
Size: 2,760,416 total tweets, 1,508 annotated tweets
Time period: 2014
Annotation: polarity, irony and sub-topic
Annotation method: crowdsourcing
URL: http://twita.dipinfo.di.unito.it/tw-swellfer

Italian Hate Speech Corpus (Sanguinetti et al., 2018b; Poletto et al., 2017) is a corpus of hate speech on social media towards migrants and ethnic minorities, in the context of the Hate Speech Monitoring Program of the University of Turin[5]. The tweets were collected according to a set of keywords: invadere (*invade*), invasione (*invasion*), basta (*enough*), fuori (*out*), comunist* (*communist**), african* (*African*), barcon* (*migrants boat**).

Name: Italian Hate Speech Corpus
Size: 236,193 total tweets, 6,965 annotated tweets
Time period: October 1st, 2016–April 25th, 2017
Annotation: hate speech, aggressiveness, offensiveness, stereotype, irony, intensity
Annotation method: crowdsourcing and experts
URL: http://twita.dipinfo.di.unito.it/ihsc

TWITTIRÒ (Cignarella et al., 2017) is a dataset of tweets overlapping with other datasets included in the University of Turin collection, on which a finer-grained annotation of irony is superimposed. The TWITTIRÒ tweets are taken from TWitterBuonaScuola, SENTIPOLC (see Section 3.2), and TWSpino (see Section 3.3).

Name: TWITTIRÒ
Size: 1,600 total tweets: 400 tweets from TWSpino, 600 from SENTIPOLC tweets, 600 tweets from TWitterBuonaScuola
Time period: 2012–2016
Annotation: fine-grained irony
Annotation method: experts
URL: http://twita.dipinfo.di.unito.it/twittiro

3.2 Shared Task Datasets

The large collection of Italian tweets of the University of Turin has been exploited in different occasions to extract datasets to organize shared tasks for the Italian community, in particular under the umbrella of the EVALITA evaluation campaign[6]. In this section, we describe such datasets.

SENTIPOLC The SENTIment POLarity Classification task was proposed in two editions of the EVALITA campaign, namely in 2014 (Basile et al., 2014) and 2016 (Barbieri et al., 2016). Both editions were organized into three different sub-tasks: subjectivity and polarity classification, and irony detection. The data for SENTIPOLC 2014 were gathered from TWITA and Senti-TUT (see Section 3.3), while for the 2016 edition the dataset was further expanded by including other data sources, such as TWitterBuonaScuola (see Section 3.1) and a subset of TWITA overlapping with the dataset used for the shared task on Named Entity Recognition and Linking in Italian Tweets (Basile et al., 2016, NEEL-it).

Name: SENTIPOLC
Size: 6,448 (SENTIPOLC 2014), 9,410 (SENTIPOLC 2016) tweets
Time period: 2012 (SENTIPOLC 2014), 2014 (SENTIPOLC 2016)
Annotation: subjectivity, polarity, irony
Annotation method: experts (SENTIPOLC 2014), crowdsourcing and experts (SENTIPOLC 2016)
URL: http://twita.dipinfo.di.unito.it/sentipolc

PoSTWITA (Bosco et al., 2016b) is the shared task on Part-of-Speech tagging of Twitter posts held at EVALITA 2016. Its content was extracted from the SENTIPOLC corpus described above. The PoSTWITA dataset consists of Italian tweets tokenized and annotated at PoS level with a tagset inspired by the Universal Dependencies scheme[7].

Name: PoSTWITA
Size: 6,738 tweets
Time period: 2012
Annotation: part of speech
Annotation method: experts
URL: http://twita.dipinfo.di.unito.it/postwita

After the task took place, the PoSTWITA corpus has been used in a new independent project on the development of a Twitter-based Italian treebank fully compliant with the Universal Dependencies, thus becoming **PoSTWITA-UD** (Sanguinetti et al., 2018a). In particular, the first core of the resource was automatically annotated by out-of-domain parsing experiments using different parsers. The output with the best results was then revised by two annotators for the final version of the resource.
PoSTWITA-UD has been made available in the official UD repository[8] since v2.1 release.

Name: PoSTWITA-UD
Size: 6,712 tweets
Time period: 2012
Annotation: dependency-based syntactic annotation
Annotation method: experts
URL: http://twita.dipinfo.di.unito.it/postwita-ud

[5] http://hatespeech.di.unito.it/
[6] http://www.evalita.it/
[7] http://universaldependencies.org/
[8] https://github.com/UniversalDependencies/UD_Italian-PoSTWITA

IronITA The irony detection task proposed for EVALITA 2018[9] consists in automatically classifying tweets according to the presence of irony (sub-task A) and sarcasm (sub-task B). Given the array of situations and topics where ironic or sarcastic devices can be used, the corpus has been created by resorting to multiple annotated sources, such as the already mentioned TWITTIRÒ, SENTIPOLC, and the Italian Hate Speech Corpus.

Name: IronITA
Size: 4,877 tweets
Time period: 2012–2016
Annotation: irony, sarcasm
Annotation method: crowdsourcing and experts
URL: http://twita.dipinfo.di.unito.it/ironita

HaSpeeDe The Hate Speech Detection task[10] at EVALITA 2018 consists in automatically annotating messages from Twitter and Facebook. The dataset proposed for the task is the result of a joint effort of two research groups on harmonizing the annotation previously applied to two different datasets: the first one is a collection of Facebook comments developed by the group from CNR-Pisa and created in 2016 (Del Vigna et al., 2017), while the other one is a subset of the Italian Hate Speech Corpus (described in Section 3.1). The annotation scheme has thus been simplified, and it only includes a binary value indicating whether hateful contents are present or not in a given tweet or Facebook comment. The task organizers created such harmonized scheme also in view of a cross-domain evaluation, with one dataset used for training and the other one for testing the system.

It is worth pointing out, however, that despite their joint use in the task, the resources are maintained separately, thus only the Twitter section of the dataset is part of TWITA.

Name: HaSpeeDe
Size: 4,000 tweets and 4,000 Facebook comments
Time period: 2016–2017 for the Twitter dataset, May 2016 for the Facebook dataset
Annotation: hate speech
Annotation method: crowdsourcing and experts for the Twitter dataset, experts for the Facebook dataset
URL: http://twita.dipinfo.di.unito.it/haspeede

3.3 Independently-collected Datasets

To complete the overview of the social media datasets, in this section we describe collections of tweets that have been compiled independently from TWITA. However, they are now hosted in the same infrastructure and therefore can be considered part of the same collection.

Senti-TUT (Bosco et al., 2013) is a dataset of Italian tweets with a focus on politics and irony. Senti-TUT includes two corpora: *TWNews* contains tweets retrieved by querying the Twitter search API with a series of hashtags related to Mario Monti (the Italian First Minister at the time); *TWSpino* contains tweets from Spinoza[11], a popular satirical Italian blog on politics.

Name: Senti-TUT
Size: 3,288 (TWNews), 1,159 tweets (TWSpino)
Time period: October 16th, 2011–February 3rd, 2012 (TWNews), July 2009–February 2012 (TWSpino)
Annotation: polarity, irony
Annotation method: experts
URL: http://twita.dipinfo.di.unito.it/senti-tut

Felicittà (Allisio et al., 2013) was a project on the development of a platform that aimed to estimate and interactively display the degree of happiness in Italian cities, based on the analysis of data from Twitter. For its evaluation, a gold corpus was created by Bosco et al. (2014), using the same annotation scheme provided for Senti-TUT.

Name: Felicittà
Size: 1,500 tweets
Time period: November 1st, 2013–July 7th, 2014
Annotation: polarity, irony
Annotation method: experts
URL: http://twita.dipinfo.di.unito.it/felicitta

ConRef-STANCE-ita (Lai et al., 2018) is a collection of tweets on the topic of the Referendum held in Italy on December 4, 2016, about a reform of the Italian Constitution. This is supposedly a highly controversial topic, chosen to highlight language features useful for the study of stance detection. The tweets were collected by searching for specific hashtags: #referendumcostituzionale (*constitutional referendum*), #iovotosi (*I vote yes*), #iovotono (*I vote no*). Subsequently, the collection was enriched by recovering the conversation chain from each retrieved tweet to its source, annotating triplets consisting in one tweet, one retweet, and one reply posted by the same user in a specific temporal window. The aim of the collection is to monitor the evolution of the stance of 248 users during the debate in four different temporal windows and also inspecting their social network.

[9] http://www.di.unito.it/~tutreeb/ironita-evalita18

[10] http://www.di.unito.it/~tutreeb/haspeede-evalita18

[11] http://www.spinoza.it

Name: ConRef-STANCE-ita
Size: 2,976 tweets (963 triplets)
Time period: November 24th, 2016–December 7th, 2016
Annotation: stance
Annotation method: crowdsourcing and experts
URL: http://twita.dipinfo.di.unito.it/conref-stance-ita

3.4 Work in Progress and Other Datasets

Finally, there are a number of additional datasets hosted in our infrastructure that are being actively developed at the time of this writing. Some of those datasets include a collection of geo-localized tweets on the 2016 edition of the "giro d'Italia" cycling competition, a dataset of tweets concerning the 2016 local elections in 10 major Italian cities, and an addendum to the ConRef-STANCE-ita dataset described in Section 3.3.

Furthermore, we limited this report to the datasets of tweets in the Italian language, which make for the majority of our collection. However, we curate several datasets in other languages, often as a result of collaborations with international research teams and projects, such as, for instance, **TwitterMariagePourTous** (Bosco et al., 2016a), a corpus of 2,872 French tweets extracted in the period 16th December 2010 - 20th July 2013 on the topic of same-sex marriage. In addition, several new corpora have been developed within the Hate Speech Monitoring program (see Section 3.1), aiming at studying hate speech phenomenon against different targets such as women and the LGBTQ community, and resorting to other data sources than Twitter (Facebook and online newspapers in particular). Although such resources are still under construction - therefore it is not possible to provide any corpus statistics yet - our goal is to include them in our resource infrastructure, thus making a step forward and ensuring its improvement also in terms of diversity of data sources.

4 Data Availability

The main goal of collecting and organizing datasets such as the ones described in this paper is, generally speaking, to provide the NLP research community with powerful tools to enhance the state of the art of language technologies. Therefore, our default policy is to share as much data as possible, as freely as possible. Twitter has proven to behave cooperatively towards the scientific community, relaxing the limits imposed to data sharing for non-commercial use over time[12]. However, there are considerations about the privacy of the users that must be accounted for in releasing Twitter data. In particular, the EU General Data Protection Regulation from 2018 (GDPR)[13] strictly regulates data and user privacy. For instance, if a tweet has been deleted by a user, it should not be published in other forms (Article 17), although it can still be used for scientific purposes.

Technically, we follow these consideration by implementing an interface to download the ID of the tweets in our collection, and tools to retrieve the original tweets (if still available). The annotated datasets can instead be shared in their entirety, given their limited size, thus we provide links to download them in tabular format. Finally, we are developing interactive interfaces to select and download samples of the collection based on the time period and sets of keywords and hashtags.

[12] https://developer.twitter.com/en/developer-terms/agreement-and-policy.html

[13] https://gdpr-info.eu/

Acknowledgments

Valerio Basile and Manuela Sanguinetti are partially supported by Progetto di Ateneo/CSP 2016 (*Immigrants, Hate and Prejudice in Social Media*, S1618_L2_BOSC_01).

Mirko Lai is partially supported by Italian Ministry of Labor (*Contro l'odio: tecnologie informatiche, percorsi formativi e story telling partecipativo per combattere l'intolleranza*, avviso n.1/2017 per il finanziamento di iniziative e progetti di rilevanza nazionale ai sensi dell'art. 72 del d.l. 3 luglio 2017, n. 117 - anno 2017).

References

Leonardo Allisio, Valeria Mussa, Cristina Bosco, Viviana Patti, and Giancarlo Ruffo. 2013. Felicittà: Visualizing and estimating happiness in Italian cities from geotagged tweets. In *Proceedings of the First International Workshop on Emotion and Sentiment in Social and Expressive Media: approaches and perspectives from AI (ESSEM 2013)*, pages 95–106, Turin, Italy.

Francesco Barbieri, Valerio Basile, Danilo Croce, Malvina Nissim, Nicole Novielli, and Viviana Patti. 2016. Overview of the Evalita 2016 SENTIment POLarity Classification Task. In *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)*, Naples, Italy.

Valerio Basile and Malvina Nissim. 2013. Sentiment analysis on Italian tweets. In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 100–107.

Valerio Basile, Andrea Bolioli, Malvina Nissim, Viviana Patti, and Paolo Rosso. 2014. Overview of the Evalita 2014 SENTIment POLarity Classification Task. In *Proceedings of the Fourth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2014)*, Pisa, Italy.

Pierpaolo Basile, Annalina Caputo, Anna Lisa Gentile, and Giuseppe Rizzo. 2016. Overview of the EVALITA 2016 Named Entity rEcognition and Linking in Italian tweets (NEEL-IT) task. In *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC-it 2016) & the Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016)*, Naples, Italy.

Cristina Bosco, Viviana Patti, and Andrea Bolioli. 2013. Developing corpora for sentiment analysis: The case of irony and Senti-TUT. *IEEE Intelligent Systems*, 28(2):55–63.

Cristina Bosco, Leonardo Allisio, Valeria Mussa, Viviana Patti, Giancarlo Ruffo, Manuela Sanguinetti, and Emilio Sulis. 2014. Detecting happiness in Italian tweets: Towards an evaluation dataset for sentiment analysis in Felicittà. In *Proceedings of the 5th International Workshop on EMOTION, SOCIAL SIGNALS, SENTIMENT & LINKED OPEN DATA*, pages 56 – 63.

Cristina Bosco, Mirko Lai, Viviana Patti, and Daniela Virone. 2016a. Tweeting and being ironic in the debate about a political reform: the French annotated corpus Twitter-MariagePourTous. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016*, Portorož, Slovenia.

Cristina Bosco, Fabio Tamburini, Andrea Bolioli, and Alessandro Mazzei. 2016b. Overview of the EVALITA 2016 Part Of Speech on TWitter for ITAlian task. In *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC-it 2016) & the Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016)*, Naples, Italy.

Alessandra Teresa Cignarella, Cristina Bosco, and Viviana Patti. 2017. Twittirò: a social media corpus with a multi-layered annotation for irony. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017)*, Rome, Italy.

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, Venice, Italy.

Mirko Lai, Viviana Patti, Giancarlo Ruffo, and Paolo Rosso. 2018. Stance evolution and twitter interactions in an italian political debate. In *NLDB*, volume 10859 of *Lecture Notes in Computer Science*, pages 15–27. Springer.

Fabio Poletto, Marco Stranisci, Manuela Sanguinetti, Viviana Patti, and Cristina Bosco. 2017. Hate speech annotation: Analysis of an Italian Twitter corpus. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017)*, Rome, Italy.

Manuela Sanguinetti, Emilio Sulis, Viviana Patti, Giancarlo Ruffo, Leonardo Allisio, Valeria Mussa, and Cristina Bosco. 2014. Developing corpora and tools for sentiment analysis: the experience of the University of Turin group. In *First Italian Conference on Computational Linguistics (CLiC-it 2014)*, pages 322–327, Pisa, Italy.

Manuela Sanguinetti, Cristina Bosco, Alberto Lavelli, Alessandro Mazzei, Oronzo Antonelli, and Fabio Tamburini. 2018a. PoSTWITA-UD: an Italian Twitter treebank in Universal Dependencies. In *Proceedings of the 11th Language Resources and Evaluation Conference LREC 2018)*, pages 1768–1775, Miyazaki, Japan.

Manuela Sanguinetti, Fabio Poletto, Cristina Bosco, Viviana Patti, and Marco Stranisci. 2018b. 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. European Language Resources Association (ELRA).

Marco Stranisci, Cristina Bosco, Delia Iraz Hernndez Faras, and Viviana Patti. 2016. Annotating sentiment and irony in the online italian political debate on #labuonascuola. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Emilio Sulis, Cristina Bosco, Viviana Patti, Mirko Lai, Delia Irazú Hernández Farías, Letizia Mencarini, Michele Mozzachiodi, and Daniele Vignoli. 2016. Subjective well-being and social media. A semantically annotated Twitter corpus on fertility and parenthood. In *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC-it 2016) & the Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016)*, Naples, Italy.

E. Tjong Kim Sang and A. van den Bosch. 2013. Dealing with big data: The case of Twitter. *Computational Linguistics in the Netherlands Journal*, 3(12/2013):121–134. Reporting year: 2013.

Neural Surface Realization for Italian

Valerio Basile
Dipartimento di Informatica
Università degli Studi di Torino
Corso Svizzera 185, 10153 Torino
basile@di.unito.it.com

Alessandro Mazzei
Dipartimento di Informatica
Università degli Studi di Torino
Corso Svizzera 185, 10153 Torino
mazzei@di.unito.it

Abstract

We present an architecture based on neural networks to generate natural language from unordered dependency trees. The task is split into the two subproblems of word *order prediction* and *morphology inflection*. We test our model gold corpus (the Italian portion of the Universal Dependency treebanks) and an automatically parsed corpus from the Web.

(Italian) Questo lavoro introduce un'architettura basata su reti neurali per generare frasi in linguaggio naturale a partire da alberi a dipendenze. Il processo è diviso nei due sottoproblemi dell'ordinamento di parole e dell'inflessione morfologica, per i quali la nostra architettura prevede due modelli indipendenti, il cui risultato è combinato nella fase finale. Abbiamo testato il modello usando un gold corpus e un silver corpus ottenuto dal Web.

1 Introduction

Natural Language Generation is the process of producing natural language utterances from an abstract representation of knowledge. As opposed to Natural Language Understanding, where the input is well-defined (typically a text or speech segment) and the output may vary in terms of complexity and scope of the analysis, in the generation process the input can take different forms and levels of abstraction, depending on the specific goals and applicative scenarios. However, the input structures for generation should be at least formally defined.

In this work we focus on the final part of the standard NLG pipeline defined by Reiter and Dale (2000), that is, *surface realization*, the task of producing natural language from formal abstract representations of sentences' meaning and syntax.

We consider the surface realization of unordered Universal Dependency (UD) trees, i.e., syntactic structures where the words of a sentence are connected by labeled directed arcs in a tree-like fashion. The labels on the arcs indicate the syntactic relation holding between each word and its dependent words (Figure 1a). We approach the surface realization task in a supervised statistical setting. In particular, we draw inspiration from Basile (2015) by dividing the task into the two independent subtasks of **word order** prediction and **morphology inflection** prediction. Two neural network-based models run in parallel on the same input structure, and their output is later combined to produce the final surface form.

A first version of the system implementing our proposed architecture (called the *DipInfo-UniTo realizer*) was submitted to the shallow track of the *Surface Realization Shared Task 2018* (Mille et al., 2018). The main research goal of this paper is to provide a critical analysis for tuning the training data and learning parameters of the DipInfo-UniTo realizer.

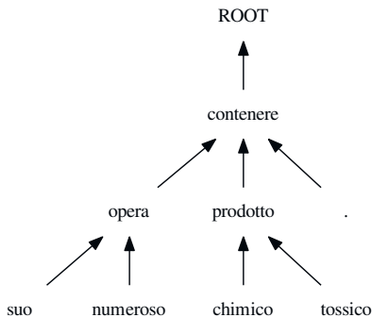
2 Neural network-based Surface Realization

In the following sections, we detail the two neural networks employed to solve the subtasks of word order prediction (2.1) and morphology inflection (2.2) respectively.

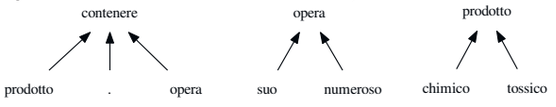
2.1 Word Ordering

We reformulate the problem of sentence-wise word ordering in terms of reordering the subtrees of its syntactical structure. The algorithm is composed of three steps: i) splitting the unordered tree into single-level unordered subtrees; ii) predicting the local word order for each subtree; iii) recomposing the single-level ordered subtrees into a single multi-level ordered tree to obtain the global word order.

In the first step, we split the original unordered universal dependency multilevel tree into a number of single-level unordered trees, where each subtree is composed by a head (the root) and all its dependents (the children), similarly to Bohnet et al. (2012). An example is shown in Figure 1:



(a) Tree corresponding to the Italian sentence "Numerose sue opere contengono prodotti chimici tossici." (*"Many of his works contain toxic chemicals."*)



(b) Three subtrees extracted from the main tree.

Figure 1: Splitting the input tree into subtrees to extract lists of items for learning to rank.

from the (unordered) tree representing the sentence "*Numerose sue opere contengono prodotti chimici tossici.*" (1a), each of its component subtrees (limited to one-level dependency) is considered separarately (1b). The head and the dependents of each subtree form an unordered list of lexical items. Crucially, we leverage the flat structure of the subtrees in order to extract structures that are suitable as input to the learning to rank algorithm in the next step of the process.

In the second step of the algorithm, we predict the relative order of the head and the dependents of each subtree with a *learning to rank* approach. We employ the list-wise learning to rank algorithm *ListNet*, proposed by Cao et al. (2007). The relatively small size of the lists of items to rank allows us to use a list-wise approach, as opposed to pair-wise or poin-twise approaches, while keeping the computation times manageable. ListNet uses a list-wise loss function based on *top one probability*, i.e., the probability of an element of being the first one in the ranking. The top one probability model approximates the *permutation probability* model that assigns a probability to each possible permutation of an ordered list. This approximation is necessary to keep the problem tractable by avoiding the exponential explosion of the number of permutations. Formally, the top one probability of an object j is defined as

$$P_s(j) = \sum_{\pi(1)=j, \pi \in \Omega_n} P_s(\pi)$$

that is, the sum of the probabilities of all the possible permutations of n objects (denoted as Ω_n) where j is the first element. $s = (s_1, ..., s_n)$ is a given list of *scores*, i.e., the position of elements in the list. Considering two permutations of the same list y and z (for instance, the predicted order and the reference order) their distance is computed using cross entropy. The distance measure and the top one probabilities of the list elements are used in the loss function:

$$L(y, z) = -\sum_{j=1}^{n} P_y(j) log(P_z(j))$$

The list-wise loss function is plugged into a linear neural network model to provide a learning environment. ListNet takes as input a sequence of ordered lists of feature vectors (the features are encoded as numeric vectors). The weights of the network are iteratively adjusted by computing a list-wise cost function that measure the distance between the reference ranking and the prediction of the model and passing its value to the gradient descent algorithm for optimization of the parameters.

The choice of features for the supervised learning to rank component is a critical point of our solution. We use several word-level features encoded as one-hot vectors, namely: the universal POS-tag, the treebank specific POS tag, the morphology features and the head-status of the word (head of the single-level tree vs. leaf). Furthermore, we included word representations, differentiating between content words and function words: for open-class word lemmas (content words) we added the corresponding language-specific word embedding to the feature vector, from the pretrained multilingual model Polyglot (Al-Rfou' et al., 2013). Closed-class word lemmas (function words) are encoded as one-hot bags of words vectors. An implementation of the feature encoding for the word ordering module of our architecture is available online[1].

[1] https://github.com/alexmazzei/ud2ln

In the third step of the word ordering algorithm, we reconstruct the global (i.e. sentence-level) order from the local order of the one-level trees under the hypothesis of projectivity[2] — see Basile and Mazzei (2018) for details on this step.

2.2 Morphology Inflection

The second component of our architecture is responsible for the morphology inflection. The task is formulated as an alignment problem between characters that can be modeled with the *sequence to sequence* paradigm. We use a deep neural network architecture based on a hard attention mechanism. The model has been recently introduced by Aharoni and Goldberg (2017). The model consists of a neural network in an encoder-decoder setting. However, at each step of the training, the model can either write a symbol to the output sequence, or move the attention pointer to the next state of the sequence. This mechanism is meant to model the natural monotonic alignment between the input and output sequences, while allowing the freedom to condition the output on the entire input sequence.

We employ all the morphological features provided by the UD annotation and the dependency relation binding the word to its head, that is, we transform the training files into a set of structures $((lemma, features), form)$ in order to learn the neural inflectional model associating a $(lemma, features)$ to the corresponding $form$. An example of training instance for our morphology inflection module is the following:

```
lemma: artificiale
features:
  uPoS=ADJ
  xPoS=A
  rel=amod
  Number=Plur
form: artificiali
```

Corresponding to the word form *artificiali*, an inflected form (plural) of the lemma *artificiale* (artificial).

3 Evaluation

In this section, we present an evaluation of the models presented in Section 2, with particular consideration for two crucial points influencing the performances of the DipInfo-UniTo realizer, namely training data and learning parameters settings. In Basile and Mazzei (2018), the hardware limitations did not allow for an extensive experimentation dedicated to the optimization of the realizer performances. In this paper, we aim to bridge this gap by experimenting with higher computing capabilities, specifically a virtualized GNU/Linux box with 16-core and 64GB of RAM.

3.1 Training Data

For our experiments, we used the four Italian corpora annotated with Universal Dependencies available on the Universal Dependency repositories[3]. In total, they comprise 270,703 tokens and 12,838 sentences. We have previously used this corpus for the training of the DipInfo-UniTo realizer that participated to the SRST18 competition (Basile and Mazzei, 2018). We refer to this corpus as *Gold-SRST18* henceforth.

Moreover, we used a larger corpus extracted from ItWaC, a large unannotated corpus of Italian (Baroni et al., 2009). We parsed ItWaC with UDpipe (Straka and Straková, 2017), and selected a random sample of 9,427 sentence (274,115 tokens). We refer to this corpus as *Silver-WaC* henceforth.

3.2 Word Ordering Performances

We trained the word order prediction module of our system[4] on the Gold-SRST18 corpus as well as on the larger corpus created by concatenating Gold-SRST18 and Silver-WaC.

The performance of the ListNet algorithm for word ordering is given in terms of average Kendall's Tau (Kendall, 1938, τ), a measure of rank correlation used to give a score to each of the rankings predicted by our model for every subtree (Figure 2). τ measures the similarity between two rankings by counting how many pairs of elements are swapped with respect to the original ordering out of all possible pairs of n elements:

$$\tau = \frac{\#concordant\_pairs - \#discordant\_tpairs}{\frac{1}{2}n(n-1)}$$

Therefore, τ ranges from -1 to 1.

In Figure 2 we reported the τ values obtained at various epochs of learning for both the Gold-

[2] As a consequence of the design of our approach, the DipInfo-UniTo realizer cannot predict the correct word order for non-projective sentences.

[3] http://universaldependencies.org/

[4] Our implementation of ListNet featuring a regularization parameter to prevent overfitting is available at https://github.com/valeriobasile/listnet

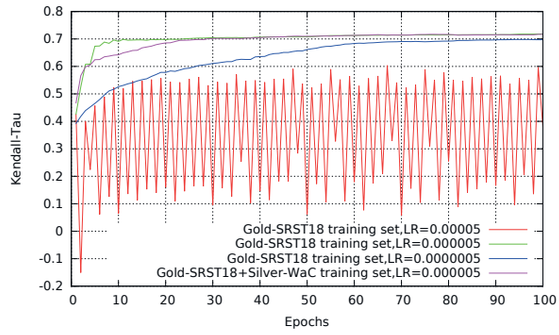


Figure 2: The trend of the τ value with respect to the ListNet iteration.

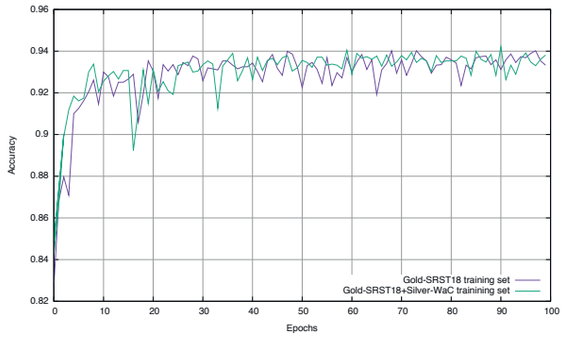


Figure 3: The trend of the Morphology Accuracy on the SRST18 development set with respect to the DNN training epochs.

SRST18 and Gold-SRST18+Silver-WaC corpora. In particular, in order to investigate the influence of the learning rate parameter (LR) in the learning of the ListNet model, we reported the τ trends for $LR = 5 \cdot 10^{-5}$ (the value originally used for the official SRST18 submission), $LR = 5 \cdot 10^{-6}$ and $LR = 5 \cdot 10^{-7}$. It is quite clear that the value of LR has a great impact on the performance of the word ordering, and that $LR = 5 \cdot 10^{-5}$ is not appropriate to reach the best performance. This explains the poor performance of the DipInfo-UniTo realizer in the SRST18 competition (Table 1). Indeed, the typical zigzag shape of the curve suggests a sort of loop in the gradient learning algorithm. In contrast, the $LR = 5 \cdot 10^{-6}$ seems to reach a plateau value after the 100th epoch with both corpora used in the experiments. We used the system tuned with this value of the learning rate to evaluate the global performance of the realizer.

3.3 Morphology Inflection Performances

In order to understand the impact of the Silver-WaC corpus on the global performance of the system, we trained the DNN system for morphology inflection[5] both on the Gold-SRST18 corpus and on the larger corpus composed by Gold-SRST18+ Silver-WaC. In Figure 3 we reported the accuracy on the SRST18 development set for both the corpora. A first analysis of the trend shows little improvement to the global performance of the realization from the inclusion of additional data (see the discussion in the next section).

3.4 Global Surface Realization Performances

Finally, we evaluate the end-to-end performance of our systems by combining the output of the two modules and submitting it to the evaluation scorer of the Surface Realization Shared Task. In Table 1 we report the performance of various tests systems with respect to the BLUE-4, DIST, NIST measures, as defined by Mille et al. (2018). The first line reports the official performance of the DipInfo-Unito realizer in the SRST18 for Italian. The last line reports the best performances achieved on Italian by the participants to SRST18 (Mille et al., 2018). The other lines report the performance of the DipInfo-UniTo realizer by considering various combination of the gold and silver corpora. The results show a clear improvement

ListNet	**Morpho**	BLEU-4	DIST	NIST
G^{srst}	G^{srst}	24.61	36.11	8.25
G	G	36.40	32.80	9.27
G	G+S	36.60	32.70	9.30
G+S	G	36.40	32.80	9.27
G+S	G+S	36.60	32.70	9.30
-	-	44.16	58.61	9.11

Table 1: The performances of the systems with respect to the BLUE-4, DIST, NIST measures.

for the word order module (note that the DIST metric is character-based, therefore it is more sensitive to the morphological variation than NIST and BLEU-4). In contrast, the morphology submodule performance seems to be unaffected by the use of a larger training corpus. This effect could be due different causes. Errors are present in the silver standard training set, and it is not clear to what extent the morphology analysis is correct

[5] An implementation of the model by (Aharoni and Goldberg, 2017) is freely available as `https://github.com/roeeaharoni/morphological-reinflection`

with respect to the syntactic analysis. The other possible cause is the neural model itself. Indeed, Aharoni and Goldberg (2017) report a plateau in performance after feeding it with relatively small datasets. The DipInfo-UniTo realizer performs better than the best systems of the SRST18 challenge for one out of three metrics (NIST).

4 Conclusion and Future Work

In this paper, we considered the problem of analysing the impact of the training data and parameters tuning on the (modular and global) performance of the DipInfo-UniTo realizer. We computationally proved that the DipInfo-UniTo realizer can gives competitive results (i) by augmenting the training data set with automatically annotated sentences, and (ii) by tuning the learning parameters of the neural models.

In future work, we intend to resolve the main lack of our approach, that is the impossibility to realize non-projective sentences. Moreover, further optimization of both neural models will be carried out on a new high-performance architecture (Aldinucci et al., 2018), by executing a systematic grid-search over the hyperparameter space, namely the regularization factor and weight initialization for ListNet, and the specific DNN hyperparameters for the morphology module.

Aknowledgment

We thank the GARR consortium which kindly allowed to use to the GARR Cloud Platform[6] to run some of the experiments described in this paper. Valerio Basile was partially funded by Progetto di Ateneo/CSP 2016 (*Immigrants, Hate and Prejudice in Social Media*, S1618_L2_BOSC_01).
Alessandro Mazzei was partially supported by the HPC4AI project, funded by the Region Piedmont POR-FESR 2014-20 programme (INFRA-P call).

References

Roee Aharoni and Yoav Goldberg. 2017. Morphological inflection generation with hard monotonic attention. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*, pages 2004–2015.

Rami Al-Rfou', Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual nlp. In *CoNLL*, pages 183–192. ACL.

[6] https://cloud.garr.it

Marco Aldinucci, Sergio Rabellino, Marco Pironti, Filippo Spiga, Paolo Viviani, Maurizio Drocco, Marco Guerzoni, Guido Boella, Marco Mellia, Paolo Margara, Idillio Drago, Roberto Marturano, Guido Marchetto, Elio Piccolo, Stefano Bagnasco, Stefano Lusso, Sara Vallero, Giuseppe Attardi, Alex Barchiesi, Alberto Colla, and Fulvio Galeazzi. 2018. Hpc4ai, an ai-on-demand federated platform endeavour. In *ACM Computing Frontiers*, Ischia, Italy, May.

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

Valerio Basile and Alessandro Mazzei. 2018. The dipinfo-unito system for srst 2018. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 65–71. Association for Computational Linguistics.

Valerio Basile. 2015. *From Logic to Language : Natural Language Generation from Logical Forms.* Ph.D. thesis, University of Groningen, Netherlands.

Bernd Bohnet, Anders Björkelund, Jonas Kuhn, Wolfgang Seeker, and Sina Zarrieß. 2012. Generating non-projective word order in statistical linearization. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 928–939. Association for Computational Linguistics.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 129–136, New York, NY, USA. ACM.

M. G. Kendall. 1938. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93.

Simon Mille, Anja Belz, Bernd Bohnet, Yvette Graham, Emily Pitler, and Leo Wanner. 2018. The first multilingual surface realisation shared task (sr'18): Overview and evaluation results. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 1–12. Association for Computational Linguistics.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

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.

Hurtlex: A Multilingual Lexicon of Words to Hurt

Elisa Bassignana and **Valerio Basile** and **Viviana Patti**
Dipartimento di Informatica
University of Turin
{basile,patti}@di.unito.it
elisa.bassignana@edu.unito.it

Abstract

English. We describe the creation of *HurtLex*, a multilingual lexicon of hate words. The starting point is the Italian hate lexicon developed by the linguist Tullio De Mauro, organized in 17 categories. It has been expanded through the link to available synset-based computational lexical resources such as MultiWordNet and BabelNet, and evolved in a multi-lingual perspective by semi-automatic translation and expert annotation. A twofold evaluation of HurtLex as a resource for hate speech detection in social media is provided: a qualitative evaluation against an Italian annotated Twitter corpus of hate against immigrants, and an extrinsic evaluation in the context of the AMI@Ibereval2018 shared task, where the resource was exploited for extracting domain-specific lexicon-based features for the supervised classification of misogyny in English and Spanish tweets.

Italiano. L'articolo descrive lo sviluppo di *Hurtlex*, un lessico multilingue di parole per ferire. Il punto di partenza è il lessico di parole d'odio italiane sviluppato dal linguista Tullio De Mauro, organizzato in 17 categorie. Il lessico è stato espanso sfruttando risorse lessicali sviluppate dalla comunità di Linguistica Computazionale come MultiWordNet e BabelNet e le sue controparti in altre lingue sono state generate semi-automaticamente con traduzione ed annotazione manuale di esperti. Viene presentata sia un'analisi qualitativa della nuova risorsa, mediante l'analisi di corpus di tweet italiani annotati per odio nei confronti dei migranti e una valutazione estrinseca, mediante l'uso della risorsa nell'ambito dello sviluppo di un sistema Automatic Misogyny Identification in tweet in spagnolo ed inglese.

1 Introduction

Communication between people is rapidly changing, in particular due to the exponential growth of the use of social media. As a privileged place for expressing opinions and feelings, social media are also used to convey expressions of hostility and *hate speech*, mirroring social and political tensions. Social media enable a wide and viral dissemination of hate messages. The extreme expressions of verbal violence and their proliferation in the network are progressively being configured as unavoidable emergencies. Therefore, the development of new linguistic resources and computational techniques for the analysis of large amounts of data becomes increasingly important, with particular emphasis on the identification of hate in language (Schmidt and Wiegand, 2017; Waseem and Hovy, 2016; Davidson et al., 2017).

The main objective of this work is the development of a lexicon of hate words that can be used as a resource to analyze and identify *hate speech* in social media texts in a multilingual perspective. The starting point is the lexicon '*Le parole per ferire*' developed by the Italian linguist Tullio De Mauro for the *"Joe Cox" Committee on intolerance, xenophobia, racism and hate phenomena* of the Italian Chamber of Deputies. The lexicon consists of more than 1,000 Italian hate words organized along different semantic categories of hate (De Mauro, 2016).

In this work, we present a computational version of the lexicon. The hate categories and lemmas have been represented in a *machine-readable* format and a semi-automatic extension and enrichment with additional information has been provided using lexical databases and ontologies. In particular we augmented the original Italian lexi-

con with translations in multiple languages.

HurtLex, the hate lexicon obtained with the method described in Section 3, has been tested with a corpus-based evaluation, through the analysis of a hate corpus of about 6,000 Italian tweets (Section 4.1), and through an extrinsic evaluation in the context of the shared task on *Automatic Misogyny Identification* at *IberEval 2018*, focusing on the identification of hate against women in Twitter in English and Spanish (Section 4.2).

The resource is available for download at `http://hatespeech.di.unito.it/resources.html`

2 Related Work

Lexical knowledge for the detection of hate speech, and abusive language in general, has received little attention in literature until recently. Even for English, there are few publicly available domain-independent resources — see for instance the novel lexicon of abusive words recently proposed by (Wiegand et al., 2018). Indeed, lexicons of abusive words are often manually compiled specifically for a task, thus they are rarely based on deep linguistic studies and reusable in the context of new classification tasks. Moreover, the lexical knowledge exploited in this context is often limited to inherently derogative words (such as slurs, swear words, taboo words). De Mauro (2016) highlights that this can be a restriction in the compilation of a lexicon of hate words, where the accent is also on derogatory epithets aimed at hurting weak and vulnerable categories of people, targeting individuals and groups of individuals on the basis of race, nationality, religion, gender or sexual orientation (Bianchi, 2014).

Regarding Italian, apart from the lexicon of hate words developed by Tullio De Mauro described in Section 3, the literature is sparse, but it is worth mentioning at least the study by Pelosi et al. (2017) on mining offensive language on social media and the project reported in D'Errico et al. (2018) on distinguishing between pro-social and anti-social attitudes. Both the works rely on the use of corpora of Facebook posts. In particular, in Pelosi et al. (2017) the focus is on automatically annotating hate speech in a corpus of posts from the Facebook page "Sesso Droga e Pastorizia", by exploiting a lexicon-based method using a dataset of Italian taboo expressions.

To conclude, let us mention that a new shared task on hate speech detection has been proposed in the context of the EVALITA 2018 evaluation campaign[1], which provides a stimulating setting for discussion on the role of lexical knowledge in the detection of hate in language.

3 Method

Our lexicon was created starting from preexisting lexical resources. In this section we give an overview of such resources and of the process we followed to create HurtLex.

3.1 "Parole per Ferire"

We started from the lexicon of "words to hurt" *Le parole per ferire* by the Italian linguist Tullio De Mauro (De Mauro, 2016). This lexicon includes more than 1,000 Italian words from 3 macro-categories: *derogatory* words (all those words that have a clearly offensive and negative value, e.g. slurs), words bearing *stereotypes* (typically hurting individuals or groups belonging to vulnerable categories) and words that are neutral, but which can be used to be derogatory in certain contexts through semantic shift (such as metaphor). The lexicon is divided into 17 finer-grained, more specific sub-categories that aim at capturing the context of each word (see also Table 1):

Negative stereotypes ethnic slurs (PS); locations and demonyms (RCI); professions and occupations (PA); physical disabilities and diversity (DDF); cognitive disabilities and diversity (DDP); moral and behavioral defects (DMC); words related to social and economic disadvantage (IS).

Hate words and slurs beyond stereotypes plants (OR); animals (AN); male genitalia (ASM); female genitalia (ASF); words related to prostitution (PR); words related to homosexuality (OM).

Other words and insults descriptive words with potential negative connotations (QAS); derogatory words (CDS); felonies and words related to crime and immoral behavior (RE); words related to the seven deadly sins of the Christian tradition (SVP).

3.2 Lexical Resources

WordNet (Fellbaum, 1998) is a lexical reference system for the English language based on psycholinguistic theories of human lexical memory.

[1] `http://www.di.unito.it/~tutreeb/haspeede-evalita18`

Category	Percentage	Category	Percentage
PS	3,85%	ASM	7,07%
RCI	0,81%	ASF	2,78%
PA	7,52%	PR	5,01%
DDF	2,06%	OM	2,78%
DDP	6,00%	QAS	7,34%
DMC	6,98%	CDS	26,68%
IS	1,52%	RE	3,31%
OR	1,52%	SVP	4.83%
AN	9,94%		

Table 1: Distribution of sub-categories in *Le parole per ferire*.

Category	Percentage	Category	Percentage
PS	2,76%	ASM	6,21%
RCI	0,41%	ASF	1,66%
PA	5,38%	PR	1,66%
DDF	1,52%	OM	2,76%
DDP	8,55%	QAS	11,03%
DMC	7,45%	CDS	26,07%
IS	1,38%	RE	4,69%
OR	2,34%	SVP	6.07%
AN	10,07%		

Table 2: Distribution of the words not present in BabelNet along the 17 sub-categories of De Mauro.

WordNet is structured around *synsets* (sets of synonyms) and their 4 coarse-grained parts of speech: noun, verb, adjective and adverb.

MultiWordNet (Pianta et al., 2002), is an extension of WordNet that contains mappings between the English lexical items in Wordnet and lexical items of other languages, including Italian.

BabelNet (Navigli and Ponzetto, 2012) is a combination of a multilingual encyclopedic dictionary and a semantic network that links concepts and named entities in a very wide network of semantic relationships.

3.3 A Computational Lexicon of Hate Words

The first step for the creation of our lexicon consisted in extracting every item from the lexicon *Le parole per ferire*. We obtain 1,138 items, but 1,082 unique items because several items were duplicated in multiple categories. We also removed 10 lemmas that belong to idiomatic multi-word-expressions, e.g., "coccodrillo" (crocodile) in the expression "lacrime di coccodrillo" (crocodile tears), leaving us to 1,072 unique lemmas.

As a second step, we use MultiWordNet to augment the words with their part-of-speech tags. We use the Italian index of MultiWordNet, comprising, for each lemma, four fields containing the identifiers of the synsets in which the lemma is intended like a noun, an adjective, a verb and a pronoun. By joining this index with our lexicon, we obtain all the possible part-of-speech for 59,2 % of the lemmas, bringing the total number of lemmas from 1,072 to 1,156 to include duplicates with different part of speech. The remaining lemmas were annotated manually.

The third step consists of linking the lemmas of the lexicon with a definition. We use the BabelNet API to retrieve the definitions, aiming for high coverage. In total, we were able to retrieve a definition for 71,1% of the lemmas. Table 2 shows the distribution of the words not present in BabelNet across the HurtLex categories. All the information about the entries of HurtLex (lemma, part of speech, definition) and the hierarchy of categories is collected in one XML structured file for distribution in machine-readable format.

3.4 Semi-automatic Multilingual Extension of the Lexicon

We leverage BabelNet to translate the lexicon into multiple languages, by querying the API[2] to retrieve all the senses of all the words in the lexicon.

Next, we queried the BabelNet API again to retrieve all the lemmas in all the supported languages, thus creating a basis for a multilingual lexicon starting from an Italian resource.

Not surprisingly, some of the senses retrieved in the first step were unrelated to the offensive context, therefore their translation to other languages would generate unlikely candidates for a lexicon of hate words. For instance, BabelNet senses of named entities which are homograph to words in the input lexicon are extracted along with the other senses, but they are typically to exclude from a resource such as HurtLex.

Therefore, we performed a manual filtering of the senses prior to the automatic translation, with the aim of translating the original words only according to their offensive meaning. We manually annotated each pair lemma-sense according to one of three classes: **Not offensive** (used for senses that are totally unrelated to any offensive context), **Neutral** (senses that are not inherently offensive, but are linked to some offensive use of the word, for example by means of a semantic shift), and **Offensive** (senses that embody a crystallized offensive use of a word). To check the consistency

[2] https://babelnet.org/guide#java

Definition	Annotation
Finocchio is a station of Line C of the Rome Metro.	Not offensive
Aromatic bulbous stem base eaten cooked or raw in salads.	Neutral[3]
Offensive term for an openly homosexual man.	Offensive

Table 3: Annotation of three senses of the Italian word "Finocchio".

Category	Occurrence	Category	Occurrence
RE	45,10%	DDP	1,90%
QAS	23,32%	IS	1,60%
CDS	8,30%	SVP	0,50%
PS	7,10%	RCI	0,30%
ASM	2,70%	PR	0,30%
OM	2,20%	DDF	0,30%
AN	2,10%	OR	0,20%
PA	2,00%	ASF	0,00%
DMC	1,90%		

Table 4: Percentage of messages in the hate speech corpus containing words from the 17 HurtLex categories.

of the annotation, a subset of 200 senses were annotated by two experts, reporting an agreement on 87.6% of the items. Table 3 shows examples of the different annotation of senses of the same word.

After discussing the results of the pilot annotation, we decided to split the *Neutral* class into two additional classes. One of the new classes covers the cases where a sense is **not literally pejorative**, but it is used to insult by means of a semantic shift, e.g. metaphorically. The other additional class is for the senses which have a clear **negative connotation**, but not necessarily a direct derogatory use in a derogatory way, e.g., the main senses of "criminal". Subsequently, the lexicon was annotated by two other experts reporting an agreement on 61% of the items. Most disagreement was concentrated in the distinctions *Not offensive*/*Not literally pejorative* (43% of the disagreement cases) and *Negative connotation*/*Offensive* (25% of the disagreement cases).

After the annotation, we discarded all the senses marked "not offensive", and created two different versions of the multilingual lexicon in 53 languages: one containing only the translations of "offensive" senses (more conservative), and the other containing translations of "offensive", "not literally pejorative" and "negative connotation" senses (more inclusive).

4 Evaluation

We evaluated the quality of the lexicon of hate words created with the method described in the previous section in two settings: by studying the occurrence of its words and their categories in a corpus of hate speech (Section 4.1), and by extracting features from HurtLex for supervised classification of misogyny in social media text (Section 4.2).

4.1 Qualitative Evaluation

In order to gain insights on the composition of the HurtLex lexicon, we evaluated it against an annotated corpus of Hate Speech on social media, recently published by Sanguinetti et al. (2018b). The corpus consists of 6,008 tweets selected according to keywords related to immigration and ethnic minorities. Each tweet in the corpus is annotated following a rich schema, including hate speech (yes/no), aggressiveness (strong/weak/none), offensiveness (strong/weak/none), irony (yes/no) and stereotype (yes/no).

We searched the lemmas of HurtLex in the version of the hate speech corpus enriched with Universal Dependencies annotations[4], by matching the pairs (lemma, POS-tag) in HurtLex with the morphosyntactic annotation of the corpus, and computed several statistics on the actual usage of such words in a specific abusive context of hate against immigrants. Table 4 shows the rate of messages in the corpus featuring words from each HurtLex category in the corpus.

For a more in-depth analysis, we also examined the relative frequency of single words in HurtLex with respect to the finer-grained annotation of the messages where they occur. Figures 1, 2, 3, 4 and 5 show examples of such analysis.

It can be noted how the relative frequency of words like "terrorismo" (*terrorism*), "ladro" (*thief*) and "rubare" (*stealing*) decrease drastically as the tweets become more aggressive, *offensive* or with a higher level of hate speech (perhaps because, albeit negative, they are not swear words)), while

[3]The derogatory use of the word "finocchio" (fennel) in Italian is thought to originate from the middle ages, linking the fennel plant to the execution of gay men at the burning stake.

[4]The corpus of hate speech by Sanguinetti et al. (2018b) has been annotated with a method similar to that described in Sanguinetti et al. (2018a).

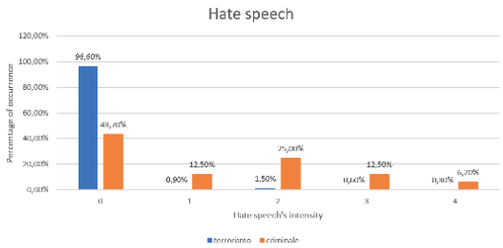


Figure 1: Relative frequency of the words "terrorismo" (*terrorism*) and "criminale" (*criminal*) with respect to the hate speech annotation.

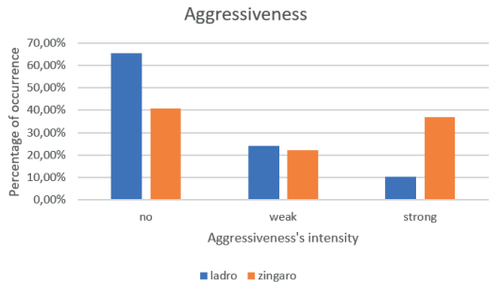


Figure 2: Relative frequency of the words "ladro" (thief) and "zingaro" (gypsy) with respect to the aggressiveness annotation.

words like "bastardo" (*bastard*) occur more as the tweets become more offensive (possibly also because they belong to the swearing sphere). Another class of words, like "zingaro" (*gypsy*), show a parabolic distribution. We hypothesize that this behavior is typical of words with an apparently neutral connotation that are sometimes used in abusive context with an offensive connotation. We plan to leverage this method of analysis for further studies on this line.

4.2 Misogyny Identification on Social Media

HurtLex was one of the resources used by the Unito's team to participate to the shared task *Automatic Misogyny Identification* (AMI) at IberEval 2018 (Pamungkas et al., 2018). The task consists of identifying misogynous content in Twitter messages (first sub-task) and classifying their misogynist behavior (second sub-task). The Unito's team employed different subsets of the 17 categories of HurtLex by extracting lexicon-based features for a supervised classifier. They identified the *Prostitution*, *Female and Male Sexual Apparatus* and *Physical and Mental Diversity and Disability* categories as the most informative for this task. The

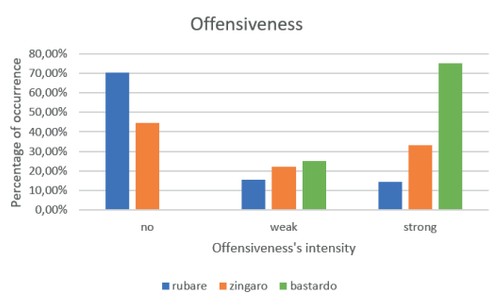


Figure 3: Relative frequency of the words "rubare" (stealing), "zingaro" (*gypsy*) and "bastardo" (bastard) with respect to the offensiveness annotation.

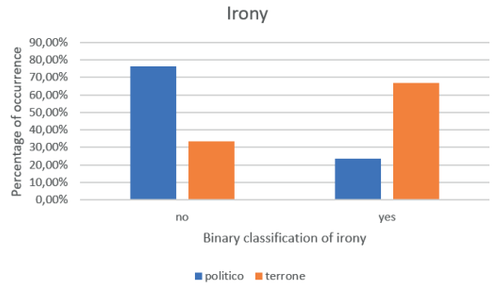


Figure 4: Relative frequency of the words "politico" (politician) and "terrone" (slur referring to southern Italians) with respect to the irony annotation.

Unito classifier obtained the best result in the first sub-task for both languages and the best result in the second sub-task for Spanish.

5 Conclusion and Future Work

Our main contribution is a machine-readable version of the hate words lexicon by De Mauro, enriched with lexical features from available computational resources. We make *HurtLex* available for download as a tool for hate speech detection. A first evaluation of the lexicon against corpora featuring different targets of hate (immigrants and women) has been presented. The multilingual evaluation of HurtLex showed also promising results. Although we are aware that hate speech-related phenomena tend to follow regional and cultural patterns, our semi-automatically produced resource was able to partially fill the gap towards hate speech detection in less represented languages. To this end, we aim at investigating the potential and pitfalls of semi-automating mappings further. In particular, two possible ex-

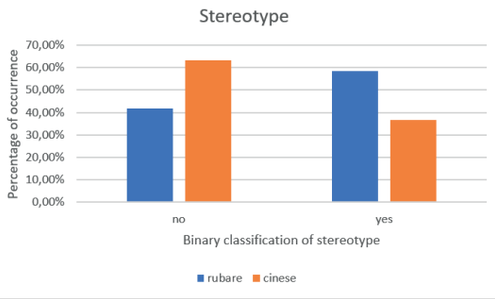


Figure 5: Relative frequency of the words "rubare" (stealing) and "cinese" (chinese) with respect to the stereotype annotation.

tensions of our method involve using distributional semantic models to automatically expand the lexicon with synonyms and lemmas semantically related to the original ones, and exploiting De Mauro's derivational rules.

Acknowledgments

Valerio Basile and Viviana Patti were partially supported by Progetto di Ateneo/CSP 2016 (*Immigrants, Hate and Prejudice in Social Media-IhatePrejudice*, S1618_L2_BOSC_01).

References

Claudia Bianchi. 2014. The speech acts account of derogatory epithets: some critical notes. In J. Dutant, D. Fassio, and Meylan A., editors, *Liber Amicorum Pascal Engel, University of Geneva*, pages pp. 465–480.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *International AAAI Conference on Web and Social Media*.

Tullio De Mauro. 2016. Le parole per ferire. *Internazionale*. 27 settembre 2016. Compiled for the "Joe Cox" Committee on intolerance, xenophobia, racism and hate phenomena, of the Italian Chamber of Deputies, which issued a Final Report in 2017.

Francesca D'Errico, Marinella Paciello, and Matteo Amadei. 2018. Prosocial words in social media discussions on hosting immigrants. insights for psychological and computational field. In *Symposium on Emotion Modelling and Detection in Social Media and Online Interaction, In conjunction with the 2018 Convention of the Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB 2018)*.

Christiane Fellbaum. 1998. *WordNet: an electronic lexical database*. MIT Press.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network. *Artificial Intelligence*, 193:217–250.

Endang Wahyu Pamungkas, Alessandra Teresa Cignarella, Valerio Basile, and Viviana Patti. 2018. 14-ExLab@UniTo for AMI at IberEval2018: Exploiting Lexical Knowledge for Detecting Misogyny in English and Spanish Tweets. In *Proc. of 3rd Workshop on Evaluation of Human Language Technologies for Iberian Languages (IberEval 2018) co-located with SEPLN 2018)*, volume 2150 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Serena Pelosi, Alessandro Maisto, Pierluigi Vitale, and Simonetta Vietri. 2017. Mining offensive language on social media. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017), Rome, Italy, December 11-13, 2017.*

Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. 2002. Multiwordnet: developing an aligned multilingual database. In *Proceedings of the First International Conference on Global WordNet*, January.

Manuela Sanguinetti, Cristina Bosco, Alberto Lavelli, Alessandro Mazzei, Oronzo Antonelli, and Fabio Tamburini. 2018a. PoSTWITA-UD: an Italian Twitter Treebank in Universal Dependencies. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 7-12, 2018. European Language Resources Association (ELRA).

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

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. Association for Computational Linguistics.

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. ACL.

Michael Wiegand, Josef Ruppenhofer, Anna Schmidt, and Clayton Greenberg. 2018. Inducing a lexicon of abusive words – a feature-based approach. 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 1046–1056. Association for Computational Linguistics.

CoreNLP-it: A UD pipeline for Italian based on Stanford CoreNLP

Alessandro Bondielli[1], **Lucia C. Passaro**[2] and **Alessandro Lenci**[2]
[1] Dipartimento di Ingegneria dell'Informazione (DINFO), Università degli studi di Firenze
[2] CoLing Lab, Dipartimento di Filologia, Letteratura e Linguistica (FiLeLi), Università di Pisa
alessandro.bondielli@unifi.it
lucia.passaro@fileli.unipi.it
alessandro.lenci@unipi.it

Abstract

English. This paper describes a collection of modules for Italian language processing based on CoreNLP and Universal Dependencies (UD). The software will be freely available for download under the GNU General Public License (GNU GPL). Given the flexibility of the framework, it is easily adaptable to new languages provided with an UD Treebank.

Italiano. *Questo lavoro descrive un insieme di strumenti di analisi linguistica per l'Italiano basati su CoreNLP e Universal Dependencies (UD). Il software sarà liberamente scaricabile sotto licenza GNU General Public License (GNU GPL). Data la sua flessibilità, il framework è facilmente adattabile ad altre lingue con una Treebank UD.*

1 Introduction

The fast-growing research field of Text Mining and Natural Language Processing (NLP) has shown important advancements in recent years. NLP tools that provide basic linguistic annotation of raw texts are a crucial building block for further research and applications. Most of these tools, like NLTK (Bird et al., 2009) and Stanford CoreNLP (Manning et al., 2014), have been developed for English, and, most importantly, are freely available. For Italian, several tools have been developed during the years such as TextPro (Pianta et al., 2008) and the Tanl Pipeline (Attardi et al., 2010) but unfortunately they are either outdated or not open source. An exception is represented by Tint (Aprosio and Moretti, 2016), a standalone freely available and customizable software based on Stanford CoreNLP. The main drawback of this solution is that it is a resource highly tailored for Italian in which some of the modules have been completely re-implemented on new classes and data structures compared to the CoreNLP ones. In addition, like for the other existing resources, it does not provide an output that is fully compatible with the Universal Dependency (UD) framework,[1] which is becoming the *de facto* standard especially for morpho-syntactic annotation, as well as for text annotation in general.

In this paper, we present CoreNLP-it, a set of customizable classes for CoreNLP designed for Italian. Our system, despite being simpler than any of the above mentioned toolkits, both in scope and number of features, has the advantage of being easily integrated with the CoreNLP suite, since its development has been grounded on the principle that all data structures be natively supported by CoreNLP.

The key properties of CoreNLP-it are:

- **UD based and compliant**: The toolkit and models are based on UD and follow its guidelines for token and parsing representation. It can provide all annotation required in the UD framework, and produces a CoNLL-U formatted output at any level of annotation, as well as any other type of annotation provided in CoreNLP.
- **Multi-word token representation**: Multi-word tokens (e.g., enclitic constructions) are handled by providing separate tokens. Moreover, the CoNLL-U output can represent such information following the UD guidelines.
- **Hybrid tokenization**: A fast and accurate hybrid tokenization and sentence splitting module replaces the original rule-based annotators for this task.
- **Integration with CoreNLP**: Given the way it is built (including the exclusive usage of

[1] http://universaldependencies.org/

CoreNLP classifiers and data structures), the add-on can be seamlessly integrated with the latest available version (3.9.1) of CoreNLP, and is expected to work with upcoming versions as well.

- **Support for other languages**: It provides out-of-the-box new capabilities of supporting basic annotations for other languages provided with a UD Treebank.

This paper is organized as follows: in Section 2, we present the architecture of the toolkit, whereas its core components (annotators) are described in Section 3. The results on Italian are discussed in Section 3.5. Section 4 shows preliminary experiments for the adaptation of the software to two additional languages provided with a UD treebank, namely Spanish and French.

2 Architecture

CoreNLP-it has been built as an *add-on* to the Stanford CoreNLP toolkit (Manning et al., 2014). CoreNLP offers a set of linguistic tools to perform core linguistic analyses of texts in English and other languages, and produces an annotated output in various formats such as CoNLL (Nivre et al., 2007), XML, Json, etc.

2.1 Stanford CoreNLP

The main architecture of CoreNLP consists of an *annotation* object as well as a sequence of *annotators* aimed at annotating texts at different levels of analysis. Starting from a raw text, each module adds a new annotation layer such as tokenizaton, PoS tagging, parsing etc. The behavior of the single annotators can be controlled via standard Java properties. Annotators can analyze text with both *rule-based* or *statistical-based* models. While rule-based models are typically language dependent, statistical based ones can be trained directly within the CoreNLP toolkit in order to improve the performance of the default models or to deal with different languages and domains.

2.2 CoreNLP-it

The main goal we pursued in developing CoreNLP-it was to keep the original CoreNLP structure and usage intact, while enabling it to deal with Italian texts in order to produce a UD-compliant and UD-complete output. More specifically, we aimed at building a system capable of providing all textual annotations required by the UD guidelines. Moreover, our system is also compatible with standard CoreNLP functions (e.g., Named Entity Recognition (NER) and Sentiment annotation). For these reasons,we implemented a series of *custom annotators* and *statistical models* for Italian. The custom annotators replace the corresponding CoreNLP annotators leaving intact the annotation structure and output of the annotators they are replacing.

For simplicity, we used only one of the UD treebanks available for Italian, namely the UD adaptation of the ISDT Italian Treebank (Bosco et al., 2013). The resource was used to build most of the new models, as well as for training standard statistical models (e.g., PoS tagging and Dependency Parsing) available in CoreNLP. More specifically, to obtain a UD-compliant output, we trained the Italian models on the *training*, *dev*, and *test* sets provided within the treebank.

The current version of CoreNLP-it can be easily integrated and configured into CoreNLP by adding the custom annotator classes and their respective models into the pipeline. Such classes and their properties can be added in a configuration file or called via the API interface. This procedure follows the standard CoreNLP documentation and guidelines for custom annotator classes. In addition, we provide a new class (resembling a CoreNLP one) for the training of the hybrid tokenization and sentence splitting. The configuration of the classifier and the required dictionaries (cf. Section 3.1) can be specified in a separate property file.

3 Modules

The annotators described in the following sections are aimed at producing a UD compliant and complete output. The following information is extracted from text: Sentences, Tokens, Universal PoS Tags, language specific PoS Tags, Lemmas, Morphological Features, and Dependency Parse Tree for each sentence.

In this section, we briefly describe each module of our linguistic pipeline, focusing on the annotators and models it implements.

3.1 Sentence Splitting and Tokenization

Sentence Splitting and Tokenization are handled by a single classifier, namely the annotator *it_tok_sent*. The process splits raw text into sen-

tences, and each sentence into tokens. Crucially, the tokenization process can deal with both single and multi-word tokens as specified by the CoNLL-U format.

Multi word tokens such as verbs with clitic pronouns (e.g., *portar-vi* "carry to you") and articulated prepositions (prep + determiner) (e.g., *della*, *di+la* "of the"), are split into their respective components. The information about the original word and its position in the sentence is however retained within each token by exploiting the *token span* and *original word* annotations.

Tokenization is usually solved with rule-based systems able to identify word and sentence boundaries, for example by identifying white spaces and full stops. However, in order to avoid encoding such set of rules, we implemented a model inspired by Evang et al. (2013). At its core, the process is driven by a hybrid model. First, it uses a character-based statistical model to recognize sentences, tokens, and clitic prepositions. Then, a rule based dictionary is used to optimize the multi-word tokens detection and splitting.

The classifier tags each character with respect to one of the following classes: i. `S`: start of a new sentence; ii. `T`: start of a new token; iii. `I`: inside of a token; iv. `O`: outside of a token; v. `C`: start of a clitic preposition inside a token (e.g. *mandar**vi***).

The classifier is a simple implementation of the maximum entropy *Column Data Classifier* available in the Stanford CoreNLP. To train the model, we used the following feature set: i. window: a window of n characters before and after the target character; ii. the case of the character; iii. the class of the previous character.

In order to deal with multi-tokens, the system allows for a full rule-based tagging of a parametric list of multi-tokens typically belonging to a strictly language dependent closed class words. In the Italian implementation, such words are *articulated prepositions* (prep + determiner). The word list to be ignored is fed to the classifier during training.

Moreover, an additional set of rules can be applied after the classification step in order to deal with possibly misclassified items. In particular, the system simply checks each token against a dictionary of multi-words and split them accordingly. In the case of Italian, we built a dictionary of *clitic verbs* (which are instead an open class) by bootstrapping the verbs in the treebank with all possible combinations of clitic pronouns. A final tagging phase was used to merge the rule-based and statistical predictions.

3.2 Part-of-Speech Tagging

The Maximum Entropy implementation of the Part-of-Speech Tagger (Toutanova et al., 2003) provided in the Stanford CoreNLP toolkit has been used to predict language dependant PoS Tags (xPoS).

In order to annotate Universal PoS (uPoS) tags, a separate annotator class, namely *upos*, has been implemented.

For what concerns the xPoS Tagger, the Maximum Entropy model was trained on the UD-ISDT Treebank. uPoS tags are instead approached with a rule based strategy. In particular, we built a mapping between xPoS and uPoS based on the UD-ISTD Treebank. The mapping is used within the annotator to assign the uPoS tag based on the predicted xPoS tag.

3.3 Lemmatization and Morphological Annotation

In order to annotate each token with its corresponding lemma and morphological features, we developed a rule-based custom annotator. The annotator exploits a parametric dictionary, to assign lemmas based on the word *form* and PoS. In particular, the dictionary contains the lemma and *UD morphological features* for n $(form, PoS)$ pairs. The form is used as the main access key to the dictionary, while PoS is used to solve ambiguity, e.g., between *amo* as "I love" or as "fishing hook". Finally, in cases of PoS ambiguity, corpus frequency is used to select the target lemma.

The dictionary can be manually built or extracted from a UD treebank. In the latter case, the provided *Vocabulary* class has methods to extract and build a serialized model of the vocabulary.

3.4 Dependency Parsing

The Neural Network Dependency Parser implemented in Stanford CoreNLP (Chen and Manning, 2014) allows models to be trained for different languages.

As for Italian, we used FastText (Joulin et al., 2016) Italian 300dim-pretrained embeddings described in Bojanowski et al. (2017). The dependency parser was trained with the default configuration provided in Stanford CoreNLP.

3.5 CoreNLP-it performances

Table 1 reports the global performances of the currently trained models. In particular, all our models were evaluated against the UD-ISDT Treebank test set.

With respect to the Tokenization, we measured the accuracy by considering the whole output of the tokenization process (i.e., the combination of the statistical classifier and rule based multi-word tokens detection). As for Lemmatization, we tested the system by predicting the lemmas for tokens in the UD-ISDT Italian test set. PoS Tagging and Dependency Parsing were tested with the system provided in CoreNLP.

Task	Tokens/sec	Results
Tok., S.Split.	17277.4	Accuracy: 99%
xPoS Tag	7575.4	F1: 0.97
Lemma	5553.1	Accuracy: 92%
Dep. Parsing	1717.8	LAS: 86.15
		UAS: 88.57

Table 1: Evaluation of CoreNLP-it modules on the UD-ISDT Treebank test set.

We must point out that one of the main shortcomings of implementing a more statistically oriented model for tokenization with respect to a rule based one is that it may underperform in the case of badly formatted or error-filled texts, which we cannot find in most Treebanks. However, we believe that such an approach could be nonetheless very useful in that it can be automatically scaled to different linguistic registers and text genres. Moreover, most typical errors could be avoided by means of data augmentation strategies and the use of more heterogeneous data for training, such as for example the PoSTWITA-UD Treebank (Sanguinetti et al., 2018).

It is important to stress that the main focus of this work was to build a framework allowing for a fast and easy implementation of UD models based on Stanford CoreNLP from a software engineering point of view. The basic pre-trained models are intended as a proof of concept, and will require further parameter tuning to increase their performance.

4 Flexibility Towards Other Languages

One of the key goals that has driven the development of CoreNLP-it is keeping the core code implementation as language independent as possible. To obtain the required linguistic knowledge, the framework exploits statistical models or external resources. On the one hand, the use of big linguistic resources to perform some of the tasks can affect the computational performances, but the system enables the construction of basic resources from the treebank used for training. On the other hand, this framework is very flexible, especially by considering tasks like tokenization and lemmatization. In particular, the system is able to produce a full UD-compliant Stanford Pipeline for languages for which an UD Treebank is available.

In order to validate this claim, we focused on two languages closely related to Italian, namely Spanish and French. We trained the respective models on the UD-adapted corpora ES-ANCORA (Taulé et al., 2008) and FR-GSD (Hernandez and Boudin, 2013). In these cases, to detect multiword tokens we exploited the information available in these corpora. It is clear that such models are intended as an interesting UD baseline, because the linguistic information they employ is not yet as optimized as the one used by the Italian models.

Since the core of the adaptation of the Stanford Pipeline to Universal Dependencies relies on the Tokenization phase, we report here the results obtained for this task. It is clear that the rest of the models (i.e., PoS tags and Parsing) can be trained simply by following the Stanford CoreNLP guidelines. Results obtained for the tokenization modules for French and Spanish are shown in Table 2.

Task	Language	Accuracy (%)
Tok., S.Split.	Spanish	99,9
	French	99,7
Lemma	Spanish	66
	French	69

Table 2: Evaluation of CoreNLP-it modules on Spanish and French.

All statistical models have similar performances with respect to Italian ones. The main differences, as expected, concern the tasks most dependent on external resources (e.g., Lemmatization). For example, we noticed a much lower recall for multiword token identification, given the exclusive use of the examples found in the training set. The approach shows very promising results especially for tokenization and sentence splitting modules which are central for all the subsequent levels of analysis

based on UD. It is clear that for PoS Tagging and Parsing further developments based on Stanford CoreNLP and language-specific resources are required to account for the specific features of each language.

5 Conclusion and Ongoing Work

In this paper, we presented CoreNLP-it, a set of add-on modules for the Stanford CoreNLP language toolkit. Our system provides basic language annotations such as sentence splitting, tokenization, PoS tagging, lemmatization and dependency parsing, and can provide a UD-compliant output. Our rule based and statistical models achieve good performances for all tasks. In addition, since the framework has been implemented as an add-on to Stanford CoreNLP, it offers the possibility of adding other new annotators, including for example the Stanford NER (Finkel et al., 2005). Moreover, first experiments on other languages have shown very good adaptation capability with very little effort.

In the near future, we plan to refine the core code by performing extensive tests to better deal with additional UD-supported languages and optimize their performances. We also plan to release the tool as well as the basic trained models for Italian. Moreover, we intend to perform data augmentation strategies to refine our models and make them able to work properly also with ill-formed or substandard text input.

References

Alessio Palmero Aprosio and Giovanni Moretti. 2016. Italy goes to Stanford: a collection of CoreNLP modules for Italian. *CoRR*.

Giuseppe Attardi, Stefano Dei Rossi, and Maria Simi. 2010. The tanl pipeline. In *LREC Workshop on WSPP*, pages 15–21, Valletta, Malta.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*.

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.

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.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of EMNLP 2014*, pages 740–750, Doha, Qatar.

Kilian Evang, Valerio Basile, Grzegorz Chrupała, and Johan Bos. 2013. Elephant: Sequence labeling for word and sentence segmentation. In *Proceedings of EMNLP 2013*, pages 1422–1426, Seattle, Washington, USA. ACL.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of ACL 2005*, ACL '05, pages 363–370, Stroudsburg, PA, USA. ACL.

Nicolas Hernandez and Florian Boudin. 2013. Construction automatique d'un large corpus libre annoté morpho-syntaxiquement en français. In *Actes de la conférence TALN-RECITAL 2013*, pages 160–173, Sables d'Olonne, France.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hervé Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *CoRR*.

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.

Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of The CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, Prague, Czech Republic. ACL.

Emanuele Pianta, Christian Girardi, and Roberto Zanoli. 2008. The TextPro tool suite. In *Proceedings of LREC 2008*, pages 2603–2607, Marrakech, Morocco. European Language Resources Association (ELRA).

Manuela Sanguinetti, Cristina Bosco, Alberto Lavelli, Alessandro Mazzei, Oronzo Antonelli, and Fabio Tamburini. 2018. Postwita-ud: an italian twitter treebank in universal dependencies. In *Proceedings of LREC 2018*.

Mariona Taulé, Maria Antònia Martí, and Marta Recasens. 2008. AnCora: Multilevel annotated corpora for catalan and spanish. In *Proceedings of LREC 2008*, pages 96–101, Marrakech, Morocco.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of NAACL 2003*, NAACL '03, pages 173–180, Stroudsburg, PA, USA. ACL.

DARC-IT: a DAtaset for Reading Comprehension in ITalian

Dominique Brunato[◇], **Martina Valeriani**[•], **Felice Dell'Orletta**[◇]
[•] University of Pisa
marti.valeriani@gmail.com
[◇]Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC–CNR)
ItaliaNLP Lab - *www.italianlp.it*
{dominique.brunato, felice.dellorletta}@ilc.cnr.it

Abstract

English. In this paper, we present DARC-IT, a new reading comprehension dataset for the Italian language aimed at identifying 'question-worthy' sentences, i.e. sentences in a text which contain information that is worth asking a question about[1]. The purpose of the corpus is twofold: to investigate the linguistic profile of question-worthy sentences and to support the development of automatic question generation systems.

Italiano. *In questo contributo, viene presentato DARC-IT, un nuovo corpus di comprensione scritta per la lingua italiana per l'identificazione delle frasi che si prestano ad essere oggetto di una domanda*[2]*. Lo scopo di questo corpus è duplice: studiare il profilo linguistico delle frasi informative e fornire un corpus di addestramento a supporto di un sistema automatico di generazione di domande di comprensione.*

1 Introduction

Reading comprehension (RC) can be defined as "the process of simultaneously extracting and constructing meaning through interaction and involvement with written language" (Snow, 2002). Such a definition emphasizes that RC is a complex human ability that can be decomposed into multiple operations, such as coreference resolution, understanding discourse relations, commonsense reasoning and reasoning across multiple sentences. In educational scenarios, student's comprehension and reasoning skills are typically assessed through a variety of tasks, going from prediction tasks (e.g. cloze test) to retellings generation and question answering, which are costly to produce and require domain expert knowledge. Given also the challenges posed by the broad diffusion of distance learning programs, such as MOOC (Massive Open Online Courses), the automatic assessment of RC is becoming a rapidly growing research field of Natural Language Processing (NLP). While much more work has been done on developing Automated Essay Scoring (AES) systems (Passonneau et al., 2017), recent studies have focused on the automatic generation of questions to be used for evaluating humans' reading and comprehension (Du and Cardie, 2017; Afzal and Mitkov, 2014). This is not a trivial task, since it assumes the ability to understand which concepts in a text are most relevant, where relevance can be here defined as the likelihood of a passage to be worth asking a question about. The availability of large and high-quality RC datasets containing questions posed by humans on a given text thus becomes a fundamental requirement to train data-driven systems able to automatically learn what makes a passage 'question-worthy'. In this regard, datasets collected for other NLP tasks, Question Answering above all, provide a valuable resource. One of the most widely used is the Stanford Question Answering Dataset (SQuAD), (Rajpurkar et al., 2016). It contains more than 100,000 questions posed by crowdworkers on a set of Wikipedia articles, in which the answer to each question is a segment of text from the corresponding reading passage. More recently, other large RC datasets have been released: it is the case of the 'TriviaQA' dataset (Joshi et al., 2017), which is intended to be more challenging than SQuaD since it contains a higher proportion of complex ques-

[1]The corpus will be made publicly available for research purposes at the following link: http://www.italianlp.it/resources/

[2]Il corpus sarà messo a disposizione liberamente per scopi di ricerca al seguente indirizzo: http://www.italianlp.it/resources/

tions, i.e. questions requiring inference over multiple sentences. The same holds for RACE (Lai et al., 2017), which is also the only one specifically designed for educational purposes. Indeed it covers multiple domains and written styles and contains questions generated by domain experts, i.e. English teachers, to assess reading and comprehension skills of L2 learners. While all these datasets are available for the English language, to our knowledge, no similar RC datasets exist for the Italian language. In this paper we introduce a new corpus for Italian specifically conceived to support research on the automatic identification of question-worthy passages. In what follows, we first describe the typology of texts it contains and the annotation process we performed on them. We then carry out a qualitative analysis based on linguistic features automatically extracted from texts with the aim of studying, on the one hand, which features mostly discriminate question-worthy sentences from other sentences and, on the other hand, whether the two classes of sentences have a different profile in terms of linguistic complexity.

2 Dataset Collection

The first step in the process of corpus construction was the selection of appropriate materials. As noted by Lai et al. (2017), a major drawback of many existing RC datasets is that they were either crowd-sourced or automatically-generated thus paying very little attention to the intended target user; this makes them less suitable to be used in real educational scenarios. To prevent these limitations, we relied on a corpus of reading comprehension tests designed by the National Institute for the Evaluation of the Education System (INVALSI), which is the Italian institution in charge of developing standardized tests for the assessment of numeracy and literacy skills of primary, middle and high school students.

To create the corpus, we focused only on tests designed to assess students' competences in the Italian language. We thus collected a total of 86 Italian tests administered between 2003 and 2013, of which 31 targeting primary school's pupils of the second, third and fifth grade, 29 targeting students of the first and third year of middle school and 26 targeting students of first, second and third grade of high school. To each text a number of questions is associated, which aim to deeply assess student's ability of reading and understanding. As documented by the last available technical report provided by the Institute[3], the INVALSI Italian test has been designed to cover seven main aspects underlying text comprehension, namely: understanding the meaning of words; identifying explicit information; inferring implicit information; detecting elements conveying cohesion and coherence in text; comprehending the meaning of a passage by integrating both implicit and explicit information; comprehending the meaning of the whole text; generating a meaningful interpretation (e.g. understanding the message, the purpose etc.). With respect to their form, questions can be either multiple-choice (typically with 3 or 4 options, see example (1)) or, more rarely, open-ended questions (example 2).

Example (1): *Dove abita il ragno del racconto?* (Where does the spider of the story live?)
A. In un albero del bosco. (On a forest tree)
B. Sopra un fiore del bosco. (Upon a forest flower)
C. In una siepe del bosco. (In a forest hedge)

Example (2): *Dopo aver letto il testo, qual è secondo te il messaggio che vuole dare l'autore?* (After reading the text, what do you think is the message the author wants to give?)

For the purpose of our study, we selected only the first type of questions, thus obtaining a total of 354 questions. Table 1 reports some statistics about the final corpus collected from the INVALSI tests.

SchoolGrade	Texts	Sentences	Questions
2nd Primary	10	195	75
4th Primary	9	205	36
5th Primary	12	427	50
1st Middle	19	513	72
3rd Middle	10	342	48
1st High	10	303	32
2nd High	7	211	18
3rd High	9	261	23
TOT	86	2457	354

Table 1: Total number of texts, total number of sentences and corresponding questions for each school grade in DARC-IT.

[3] http://www.invalsi.it/invalsi/doc_eventi/2017/Rapporto_tecnico_SNV_2017.pdf

2.1 Annotation Scheme

For each question of the corpus, the annotation process was meant to identify the sentence (or a sentence span) containing the corresponding answer. This information was marked on text by enclosing the relevant text span in opening and closing *xml* tags with a letter R in upper case.

The outcome of the annotation process was a tabular file with the following information reported in separate columns: i) the text segmented into sentences; ii) a binary value 1 vs 0 (1 if the sentence contains the answer to the question and 0 if not); iii) the corresponding question; iv) the answer provided by the annotator. Table 2 gives an example of the dataset structure.

A qualitative inspection of the corpus allowed identifying different typologies of 'question-worthy' sentences: sentences that were the target of one question only (this is the case of the second sentence reported in Table 2); sentences that were the target of multiple questions, such as (4), and sentences that only partially answered the question (i.e. the whole information required to give the answer is spread across multiple sentences), such as (5).

(4) Question-worthy sentence: *Leo decide di aiutare gli animali della giungla* (*Leo decided to help the jungle animals*)

Corresponding questions:

- Qual è la cosa più importante per Leo? (What is the most important think to Leo?)

  Multiple choice answers: A. Essere un bravo cacciatore. (To be a good hunter); B. Diventare il piú coraggioso di tutti. (To become the bravest of all); C. Rendersi utile agli altri. (To make himself useful to others); D. Fare nuove esperienze. (To make new experiences).

- Cosa sceglie di fare Leo nella giungla? (What does Leo choose to do in the jungle?)

  Multiple choice answers: A. Giocare con tutti. (To play with everybody); B. Dormire e mangiare. (To sleep and eat); C. Aiutare chi è in difficoltà. (To help people in need); D. Nuotare nell'acqua del fiume (To swim in the river water)

(5) Question-worthy sentences: *"Io farò il postino!" Disse uno. "Io farò il maestro!" Disse un altro. "E io farò lo chef!". Urlò un terzo e salì sul vagone delle marmellate.* (I'm going to be a postman! One said. I'm going to be a teacher! Another said. And I'm going to be a chef! Shouted a third one and went up on the wagon of the jams).

Corresponding question: A che cosa pensano i bambini quando vedono gli oggetti sul treno? (What do children think when they see the items on the train?)

Multiple choice answers: A. Ai giochi che potranno fare. (To the plays they can do); B. A cose utili che si possono vendere. (To useful things that can be sold); C. Ai regali che vorrebbero ricevere. (To the presents they would like to receive); D. Ai lavori che faranno da grandi. (To the trades they will do as adults.)

3 Linguistic Analysis

As a result of the annotation process, we obtained 398 'question-worthy' sentences and 2059 'non-question' worthy sentences. Starting from this classification we carried out an in-depth linguistic analysis based on a wide set of features capturing properties of a sentence at lexical, morpho–syntactic and syntactic level. The aim of this analysis was to understand whether there are some linguistic features that mostly allow predicting the 'likelihood' of a sentence to be the target of a question. To allow the extraction of linguistic features, all sentences were automatically tagged by the part-of-speech tagger described in (Dell'Orletta, 2009) and dependency parsed by the DeSR parser described in (Attardi et al., 2009).

Table 3 shows an excerpt of the first 20 features (of 177 extracted ones) for which the average difference between their value in the 'question-worthy' and 'non question-worthy' class was highly statistically significant using the Wilcoxon rank sum test[4]. As it can be seen, sentences on which a comprehension question was asked are on average much more longer. This could be expected since the longer the sentence the higher the probability that it is more informative and thus containing concepts that are worth asking a question about. This is also suggested by the higher distribution of proper nouns [10], most likely referring to relevant semantic types (e.g. person, location) which typically occur in Narrative, i.e. the main textual genre of the Invalsi tests. The higher sentence length of 'question-worthy' sentences has effects also at morpho-syntactic and

[4] All significant features are shown in Appendix (A).

Sentence	Class	Tag	Question	Answer
La lucciola si preparò e, quando calò la sera, andò all'appuntamento.	0			
Entrò nel bosco scuro e raggiunse la siepe dove viveva il ragno.	1	Entrò <R>nel bosco scuro e raggiunse la siepe dove viveva il ragno.<\R>	Dove abita il ragno del racconto?	In una siepe del bosco.

Table 2: Sample output of the dataset structure.

syntactic level, as shown e.g. by the higher proportion of conjunctions introducing subordinate clauses ([7] *Subord. conj*: 1.63 vs 1.50) and by the presence of longer syntactic relations in which the linear distance between the 'head' and the 'dependent' is higher than 10 tokens ([20] *Max link*: 11.30 vs 8.30).

	Question		**NoQuestion**	
Features	**Avg**	**(StDev)**	**Avg**	**(StDev)**
Raw Text features				
[1] Sentence length*	29.00	(16.11)	20.00	(13.75)
Morpho–syntactic features				
[2] Punctuation*	4.74	(2.82)	7.70	(6.23)
[3] Negative adv*	1.23	(2.82)	1.19	(3.13)
[4] Coord. conj*	3.50	(3.40)	3.20	(3.81)
[5] Poss. adj*	0.96	(2.10)	0.89	(2.33)
[6] Relative pron*	1.14	(2.00)	1.12	(2.32)
[7] Subord. conj*	1.63	(2.80)	1.50	(2.90)
[8] Prepositions*	7.90	(5.01)	7.60	(6.20)
[9] Determiners*	9.13	(5.00)	9.00	(6.20)
[10] Proper nouns*	2.05	(3.90)	2.00	(4.30)
[11] Numbers	0.66	(1.87)	0.64	(2.25)
[12] Verbs	15.98	(6.32)	16.97	(8.18)
[13] Indicat. mood*	57.00	(30.70)	60.00	(33.82)
[14] Particip. mood	7.13	(14.22)	6.34	(14.88)
[15] 3rdpers. verb*	55.15	(39.50)	45.20	(42.62)
[16] Conjunctions	5.1	(4.35)	4.34	(4.66)
Syntactic features				
[17] Clause length*	8.63	(4.34)	7.90	(4.24)
[18] Verbal heads*	4.00	(2.30)	3.00	(2.03)
[19] Postverb Subj*	13.60	(27.00)	15.70	(32.00)
[20] Max link*	11.30	(7.06)	8.30	(6.80)

Table 3: Linguistic features whose average difference between the two classes was statistically significant. For each feature it is reported the average value (avg) and the standard deviation (StDev). All differences are statistically significant at p<.005; those with * also at p<.001. (Note: Question=question-worthy sent.; NoQuestion=Non question-worthy sent.)

A further analysis was meant to investigate the profile of question-worthy sentences with respect to linguistic complexity. To this end, we exploit READ-IT (Dell'Orletta et al., 2011), a general-purpose readability assessment tool for Italian, which combines traditional raw text features with lexical, morpho-syntactic and syntactic information to operationalize multiple phenomena of text complexity. READ–IT assigns different readability scores using the following four models: 1) Base Model, relying on raw text features only (e.g. average sentence and word length); 2) Lexical Model, relying on a combination of raw text and lexical features; 3) Syntax Model, relying on morpho-syntactic and syntactic features; 4) Global Model, combining all feature types (raw text, lexical, morpho-syntactic and syntactic features).

Results are reported in Table 4. As it can be noted, question-worthy sentences have a higher complexity with respect to all models. Especially at syntactic level, this could be expected given the higher values obtained by features related to syntactic complexity which turned out to be significantly involved in discriminating these sentences.

	Question	NoQuestion
READ-IT Base	59,9%	21,1%
READ-IT Lexical	98,9 %	66,4%
READ-IT Syntactic	69,3%	37,5%
READ-IT Global	100%	95%

Table 4: Readability score obtained by different READ-IT models.

4 Conclusion

We presented DARC-IT, a new reading comprehension dataset for Italian collected from a sample of standardized evaluation tests used to assess students' reading and comprehension at different grade levels. For each text, we annotated 'question-worthy' sentences, i.e. sentences which contained the answer to a given question. A qualitative analysis of these sentences showed that the likelihood of a sentence to be 'question-worthy' can be modeled using a set of linguistic features, which are especially linked to syntactic complexity. We believe that this corpus can support research on the development of automatic question generation systems as well as question answering systems. Current developments go into several directions: we are carrying out a first

classification experiment to automatically predict 'question-worthy' sentences and evaluate the impact of linguistic features on the classifier performance. We are also planning to enlarge the corpus and to investigate more in-depth the typology of questions and answers it contains, in order to study what characterizes sentences answering, for instance, to factual vs non-factual questions.

5 Acknowledgments

The work presented in this paper was partially supported by the 2–year project (2018-2020) SchoolChain – Soluzioni innovative per la creazione, la certificazione, il riuso e la condivisione di unità didattiche digitali all'interno del sistema Scuola, funded by Regione Toscana (BANDO POR FESR 2014-2020).

References

Naveed Afzal and Ruslan Mitkov. 2014. Automatic generation of multiple choice questions using dependency-based semantic relations *Soft Computing*, 18 (7), 1269–1281.

Giuseppe Attardi, Felice Dell'Orletta, Maria Simi, Joseph Turian. 2009. Accurate dependency parsing with a stacked multilayer perceptron. In *Proceedings of EVALITA 2009 - Evaluation of NLP and Speech Tools for Italian 2009*, Reggio Emilia, Italy, December 2009.

Felice Dell'Orletta. 2009. Ensemble system for part-of-speech tagging. In *Proceedings of EVALITA 2009 - Evaluation of NLP and Speech Tools for Italian 2009*, Reggio Emilia, Italy, December 2009.

Felice Dell'Orletta, Simonetta Montemagni and Giulia Venturi. 2011. READ-IT: assessing readability of Italian texts with a view to text simplification. *Proceedings of the Second Workshop on Speech and Language Processing for Assistive Technologies (SLPAT 2011)*, Edimburgo, UK: 73–83.

Xinya Du and Claire Cardie. 2017. Identifying Where to Focus in Reading Comprehension for Neural Question Generation. *In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Copenhagen, Denmark.

Mandar Joshi, Eunsol Choi, Daniel Weld and Luke Zettlemoyer. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, 1601–1611.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, Eduard H. Hovy. 2017. RACE: Large-scale ReAding Comprehension Dataset From Examinations. *In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Copenhagen, Denmark.

Rebecca J. Passonneau, Ananya Poddar, Gaurav Gite, Alisa Krivokapic, Qian Yang and Dolores Perin. 2016. Wise Crowd Content Assessment and Educational Rubrics. *International Journal of Artificial Intelligence in Education*, 28, 29–55.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Austin, Texas, pages 2383–2392.

Catherine Snow. 2002. *Reading for understanding: Toward an RD program in reading comprehension*. Rand Corporation.

Appendix (A).

	Question-worthy sentences		Non Question-worthy Sentences	
Features	**Average**	**(StDev)**	**Average**	**(StDev)**
Raw Text features				
Sentence length***	29.00	(16.11)	20.00	(13.75)
Lexical features				
% Basic Italian Vocabulary (BIV)*	88.54	(8.53)	88.99	(10.66)
% Fundamental BIV**	78.26	(10.83)	79.59	(13.23)
% 'High Usage' BIV*	12.31	(8.12)	12.50	(10.28)
Lexical density*	0.56	(0.08)	0.58	(0.11)
Morpho–syntactic features				
% Adjectives*	5.20	(4.71)	4.35	(5.55)
% Articles***	9.13	(5.00)	9.00	(6.20)
% Conjunctions**	5.1	(4.35)	4.34	(4.66)
% Coordinat. conj***	3.50	(3.40)	3.20	(3.81)
% Demonstrative determiners***	0.61	(1.61)	0.55	(1.90)
% Indefinite pronouns	0.87	(2.26)	0.66	(2.24)
% Interrogative determiners*	00.5	(0.52)	0.06	(0.67)
% Interjections*	0.03	(0.31)	0.09	(0.72)
% Numbers**	0.66	(1.87)	0.64	(2.25)
% Negative adverbs***	1.23	(2.82)	1.19	(3.13)
% Ordinal numbers*	0.27	(1.04)	0.14	(0.83)
% Possessive adjectives***	0.96	(2.10)	0.89	(2.33)
% Prepositions***	7.90	(5.01)	7.60	(6.20)
% Proper nouns**	2.05	(3.90)	2.00	(4.30)
% Punctuation***	4.74	(2.82)	7.70	(6.23)
% Relative pronouns***	1.14	(2.00)	1.12	(2.32)
% Subordin. conj***	1.63	(2.80)	1.50	(2.90)
% Verbs**	15.98	(6.32)	16.97	(8.18)
% Verb_Participial mood**	7.13	(14.22)	6.34	(14.88)
% Verb_Indicative mood***	57.00	(30.70)	60.00	(33.82)
% Verb_Conditional mood**	1.37	(6.13)	2.35	(9.58)
% Verb_Past tense**	22.19	(34.80)	23.88	(37.73)
% Verb_Imperfect tense**	29.08	(39.35)	29.04	(41.13)
% Verb_Present tense*	45.04	(43.50)	38.40	(44.91)
% 3rdpers. verb***	55.15	(39.50)	45.20	(42.62)
% 2ndpers. verb*	1.37	(7.34)	1.84	(10.25)
TTR ratio (first 100 lemmas)**	0.84	(0.10)	0.89	(0.10)
Syntactic features				
Clause length (in tokens)***	8.63	(4.34)	7.90	(4.24)
Avg verbal heads/sentence***	4.00	(2.30)	3.00	(2.03)
Avg prep. links length*	1.11	(0.45)	0.93	(0.58)
Max link length***	11.30	(7.06)	8.30	(6.80)
Verb arity	34.93	(29.74)	33.37	(32.70)
% Postverbal subject***	13.60	(27.00)	15.70	(32.00)
% Preverbal objects*	10.17	(25.17)	9.22	(25.55)
% DEP Root**	5.52	(3.31)	8.20	(6.30)
% DEP Mod_rel***	1.50	(2.21)	1.30	(2.50)
% DEP Copulative Conj**	5.34	(4.92)	4.65	(5.26)
% DEP Determiner***	9.10	(5.00)	8.80	(6.20)
% DEP Disjuntive Conj	0.14	(0.76)	0.20	(0.99)
% DEP Locative Compl*	0.73	(2.03)	0.53	(1.81)
% DEP_neg***	1.20	(2.80)	1.13	(2.84)
% DEP conj**	4.58	(4.12)	3.91	(4.62)
% DEP concatenation*	0.06	(0.52)	0.08	(0.8)

Table 5: Linguistic features whose average difference between the two classes was statistically significant. For each feature it is reported the average value and the standard deviation (StDev). *** indicates a highly significant difference ($p<.001$); ** a very significant difference ($p<.01$); * a significant difference ($p<.05$).

Modelling Italian construction flexibility with distributional semantics: Are constructions enough?

Lucia Busso **Ludovica Pannitto** **Alessandro Lenci**
CoLing Lab, University of Pisa
{lucia.busso90,ellepannitto}@gmail.com, alessandro.lenci@unipi.it

Abstract

English. The present study combines psycholinguistic evidence on Italian valency coercion and a distributional analysis. The paper suggests that distributional properties can provide useful insights on how general abstract constructions influence the resolution of coercion effects. However, complete understanding of the processing and recognition of coercion requires to take into consideration the complex intertwining of lexical verb and abstract constructions.

Italiano. *Il lavoro unisce uno studio psicolinguistico sul fenomeno della coercion valenziale in Italiano con un'analisi distribuzionale.L'articolo suggerisce che le proprietà distribuzionali forniscano un'utile passaggio per capire l'influenza delle costruzioni alla risoluzione di effetti di coercion. Tuttavia, una piena comprensione del fenomeno richiede di prendere in considerazione la complessa relazione tra verbo e costruzione argomentale.*

1 Introduction

In Construction Grammar (Goldberg, 2006), the basic units of linguistic analysis are called *constructions* (Cxns), form-meaning pairings associated with autonomous, non-compositional abstract meanings, independently from the lexical items occurring in them. Examples of Cxns range from morphemes (e.g., *pre-, -ing*), to filled or partially-filled complex words (e.g., *daredevil*) to idioms (e.g., *give the devil his dues*) to more abstract patterns like the Ditransitive [Subj V Obj1 Obj2] (e.g., *he gave her a book*) (Goldberg, 2006).

Cxns appear at any level of linguistic analysis, but the level at which the notion of constructional meaning represents a radical departure from other theories of grammar is *argument structure*. These Cxns, such as the English Ditransitive, are claimed to be associated with an abstract semantic content. In this case, constructional meaning can be paraphrased as *X CAUSES Y TO RECEIVE Z.* One of the main supporting arguments in favour of constructions as independent and primitive objects of grammar is the flexibility with which argument Cxns and verbs interact with each other, as in example (1) in which the original intransitive sense of "to sneeze" is overridden by the Caused Motion Cxn, and thus takes a transitive sense of "making something move by sneezing".

(1) *John sneezed the napkin off the table*

This flexibility in combining Cxns and verbs is known as *valency coercion* (Michaelis, 2004; Boas, 2011; Lauwers and Willems, 2011; Perek and Hilpert, 2014).

This phenomenon, although vastly addressed for English, has not yet received a systematic investigation in other languages. For notable exceptions, see Boas and Gonzálvez-García (2014). In particular – to the best of our knowledge – no previous attempt to carry out an empirical investigation of valency coercion exists for Italian. However, even a simple corpus query reveals that the phenomenon is present in Italian, though it is not as pervasive as in English:

(2) *Tossì una risata leggera tra i suoi capelli*
(He coughed a light laugh in her hair) [ItWac]

This paper presents an analysis of Italian constructional flexibility that combines psycholinguistic and computational evidence: first, we present the results of a behavioral experiment on valency coercion. Then, we model Cxns with distributional semantics to investigate whether the semantic shape of Italian argument Cxns can affect the interpretation and processing of coerced sentences.

2 Studying valency coercion: an acceptability rating task

MATERIALS AND SUBJECTS: The offline psycholinguistic experiment targets 9 Italian Cxns (see Table 1) that were selected using existing resources: *LexIt* (Lenci et al., 2012) and *ValPal* (Cennamo and Fabrizio, 2013). The resultant Cxns are of varying abstractness and schematicity levels (Barðdal, 2008).

Cxn	frames
CAUSED MOTION (CM)	NPj-V-NP -PPlocation
CAUSED MOTION + via (CMvia)	NPs-V-NPobj
DATIVE (DT)	NPs-V-NPj-PPrecipient
INTRANSITIVE MOTION (IM)	NPs-V–PPlocation
PASSIVE (PASS)	NPs-V-PP
PREDICATIVE (PRED)	NPs-V–AdjPpredicate
VERBA DICENDI explicit (sentential) (VDE)	NPs-V-cheVP
VERBA DICENDI implicit (sentential) (VDI)	NP-V-diVP

Table 1: Constructions used in the test.

For each Cxn, we built 21 sentences, which were subdivided into 3 experimental conditions: GRAMMATICAL (3a), COERCION (3b), IMPOSSIBLE (3c) (7 sentences per condition). The total number of stimuli amounts to 189 sentences. The structure of the test was inspired by Perek and Hilpert (2014). Between conditions, sentences differ only for their main verb, to have as little variation as possible.

(3) a. *Gianni ha detto che verrà domani* (Gianni said that he will come tomorrow)

b. *Gianni ha fischiettato che verrà domani* (Gianni whistled that he will come tomorrow)

c. *Gianni ha cucinato che verrà domani* (Gianni cooked that he will come tomorrow)

The coercion condition consists of verbs that display a partial semantic incompatibility with the constructional environment they are embedded in. They were selected by means of both native intuition and corpus query, selecting and refining cases that were either hapax or rare occurrences in the Italian corpus *ItWac* (Baroni et al., 2009).

120 Italian native speakers were tested: 39 adolescents (12-14 years old), 40 young adults (18-35 years old), and 41 adults (over 40). We tested subjects of different ages following extensive sociolinguistic literature that has shown that language use changes with age (Eckert, 2017; Labov, 2001; Wagner, 2012). Thus, it could be the case that grammaticality judgments on creative, non-standard sentences are also affected by age. Including different age groups in our analysis allows us to investigate a more representative sample of the population. To control for the possible influencing factor of education level, we only tested adult speakers either in possess of (at least) a bachelor degree or enrolled in a University course. Table 2 summarizes number, age groups and distribution of tested subjects.

Age group	Age range	distribution	Gender	Tot
Adolescents	12-14	mean: 12.9 sd:0.63	24 m (61,5%) 15 f (38,4%)	39
Young Adults	18-39	mean:27.3 sd:2.94	15 m (37,5%) 25 f (62,5%)	41
Adults	Over 40	mean: 56.7 sd:9.48	18 m (43,9%) 23 f (56,1%)	40

Table 2: Data about tested subjects.

A within-subject design was used, in which each subject sees all stimuli. Participants were asked to judge the acceptability of the (randomized) stimuli on a Likert scale from 1 - "completely unnatural" - to 7 - "perfectly natural". Presentation of the data varied across age groups: adolescents were given the test directly in their class. Young adults' judgments were collected through the online platform Figure Eight. Older adults, instead, were presented with a simple Microsoft Word document, in order to include participants who did not have familiarity with online data gathering.

RESULTS: We assessed statistical significance via linear mixed effect modelling, with by-subject and by-item intercepts.[1] Results show that coercion sentences (purple boxplot in Figure 1) are recognized as an intermediate condition between complete grammaticality and total ungrammaticality.[2] We consider this result to support the claim that coercion effects include a degree of semantic incompatibility that is nonetheless resolved in the interpretation process. Consistently

[1] model selection performed automatically via LRT with the R package *afex*. Models were performed with the R package *lmerTest* and R2 values were calculated with the MuMIn package (Singmann et al., 2016; Kuznetsova et al., 2017; Bartoń, 2013)

[2] $p < 0.0001$, R2c 0.61

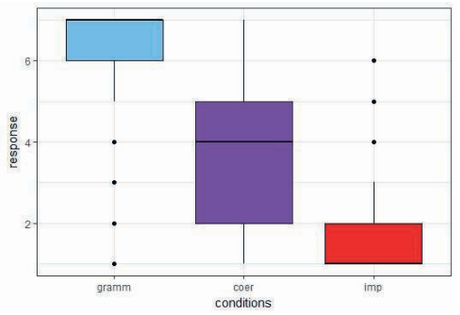


Figure 1: distribution of judgments in the 3 conditions

with the main tenets of Construction Grammar, we argue that the resolution of such incompatibility is driven by a dynamic interaction between the main verb and the constructional context (Kemmer, 2008; Kemmer and Yoon, 2013; Yoon, 2016). In a second analysis, we wanted to assess the effect of Cxn types on acceptability ratings. We used linear mixed effect modelling, adding an interaction between Cxn type and experimental condition.[3] Results indicate high variability in Cxn 'coercibility' (see Figure 2 and table 3). That is, some Cxns in our dataset were consistently judged as more natural by speakers in the coercion condition.

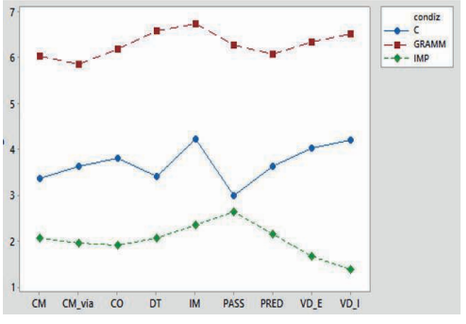


Figure 2: line plot of judgments

In particular, it appears that IM, VDE and VDI Cxns result to be more natural, while DT, PASS and (marginally) CO are the least naturally perceived ones in coercion sentences. Since coercion effects are said to be resolved by the general Cxn semantics overriding the lexical meaning of the verb, we hypothesize that the different flexibility degrees of the Cxns in the first experiment could be at least partially explained by distributional properties, such as type and token frequency, and semantic density of the Cxns in our

[3] $p < 0.0001$, R2c 0.76

	Estimate	Std. Error	t value	p value
coer	**3,64*** **	0,1	37,45	<0.0001
gramm	**2,66*** **	0,02	110,87	<0.0001
imp	**-1,79*** **	0,02	-74,84	<0.0001
CM	-0,14	0,16	-0,91	0,36
CMvia	-0,24	0,16	-1,53	0,13
CO	**-0,26.**	0,13	-1,95	0,05
DT	**-1,34*** **	0,17	-7,98	<0.0001
IM	**1,02*** **	0,16	6,40	<0.0001
PASS	**-0,73** **	0,26	-2,75	0,009
PRED	-0,07	0,26	-0,27	0,79
VDE	**1.06*** **	0,16	6,67	<0.0001
VDI	**0,70*** **	0,15	4,57	<0.0001

Table 3: fixed effects estimates of the coercion condition

dataset, the latter again estimated with distributional semantics.

Different degrees of flexibility could derive either from cognitive processes that reflect on language use, or emerge from repeated exposure and thus entrench in speakers' grammar. Both possible directions of this causal circle, however, ultimately allow us to fruitfully investigate construction flexibility using distributional semantics models. In other words, the higher 'coercibility' of novel instances of some Cxns could be due to speakers' sensitivity to distributional semantic features of the constructions (Barddal, 2006; Bybee, 2013; Zeschel, 2012; Perek and Goldberg, 2017).

3 A Distributional Semantic Model for argument constructions

PROCEDURE: Perek (2016) has shown that distributional semantics (Lenci, 2018) can be fruitfully used to model the semantic space covered by a Cxn. It has been argued in the literature that constructional meanings for argument Cxns arise from the meaning of high frequency verbs that co-occur with them (Goldberg, 1999; Casenhiser and Goldberg, 2005; Barak and Goldberg, 2017). Therefore, we modelled the semantic content of Cxns with the semantics of their most typical verb, each represented as a distributional vector.

We used the UDLex Pipeline[4] (Rambelli et al., 2017) to obtain a mapping between the Cxns of our dataset and the most frequent verbs that occur in them (these were selected considering verbs that appear at least 5 times in the relevant subcatego-

[4] The UDLex Italian dataset consist of 409,127 tokens.

rization frames). Table 4 summarizes the number of verbs considered for each of the eight Cxns.[5] Then, we built a Distributional Semantic Model (DSM) from the italian corpus *itWac* (Baroni et al., 2009) in order to represent verb meaning of the verbs obtained with UDLex. The 300-dimensional vectors (i.e., the embeddings) were created with the SGNS algorithm (Mikolov et al., 2013), using the most frequent 30,000 words as context, with a minimum frequency of 100.

Cxn	type freq (different verbs)	token freq (number of items)
CM	103	1538
CO	5	43
DT	90	1659
IM	51	1097
PASS	8	49
PRED	19	359
VD_E	12	116
VD_I	15	199

Table 4: Number of selected verbs per Cxn.

Following Lebani and Lenci (2017), we represented each Cxn as the weighted centroid vector of its typical verbs, as follows:

$$\overrightarrow{CXN} = \frac{1}{|V|} \sum v \in V frel(v, Cxn) \cdot \vec{v} \quad (1)$$

where V the set of the top-associated verbs v with Cxn and $frel(v, Cxn)$ is the co-occurrence frequency of a verb in a Cxn.

We measured the pairwise cosine similarity among the weighted Cxn vectors: as shown in Figure 3, the distributional behaviour of the Cxn vectors suggests that some Cxns in our dataset show similar distributional behaviour.

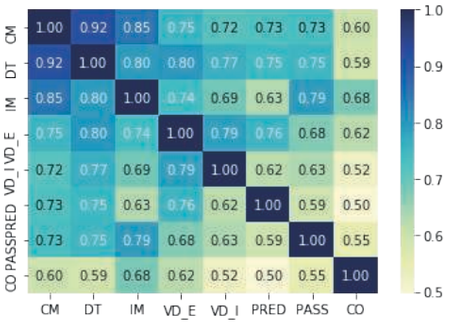


Figure 3: Construction semantic similarity.

[5]the Cxn CMvia was excluded due to the absence of corresponding subcategorization frames

Following Perek (2016), the semantic density of a Cxn is computed as the mean value of pairwise cosines between the verbs occurring in Cxn. Figure 4 plots the semantic densities of our Cxns.

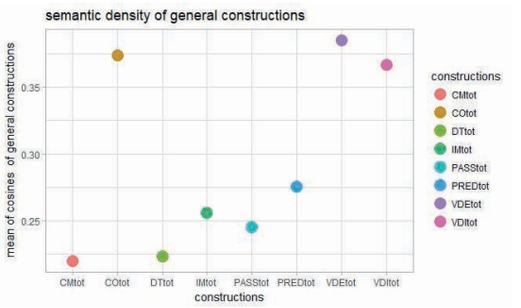


Figure 4: Construction semantic density.

Finally, to assess the effect of distributional properties on Cxns flexibility, we used semantic density, type frequency and token frequency (cf. Table 4) as predictors in linear mixed effect modelling. As dependent variable, we used the difference $gramm - coer$ and $coer - imp$. We performed two separate analyses for type and token frequencies without interactions to avoid multicollinearity effects. Predictors values were centered.

RESULTS: The estimates are reported in Tables 5 and 6 below. In the first two models frequency does not yield any effect. In the second models, instead, frequency appears to have an effect on the data. Hence, it appears that type and token frequency help discerning impossible from coercion instances of a Cxn, whereas only semantic density affects the higher naturalness of coercion phenomena. The more a Cxn is observed with semantically similar verbs (i.e., verbs that belong to the same classes or subclasses, which therefore increase the Cxn semantic density), the more the constructional meaning is easily coerced into novel instances.

4 Discussion

These findings support our claim that coercion effects are resolved by a dynamic interrelation between verb and Cxn (Kemmer, 2008; Kemmer and Yoon, 2013). Even though frequency effects are shown to affect Cxns extensibility to new items (Bybee, 2006), our results suggest that type and token frequency only facilitate the distinc-

(Gramm - coer) ∼sem. dens + type freq.				
	estimate	st. error	t value	p value
(Intercept)	**2.71*** **	0.11	25.02	<0.0001
Sem. density	**-0.34.**	0.16	-2.217	0.007
Type freq.	-0.13	0.16	-0.848	0.44
(Gramm - coer) ∼sem. dens + tok freq.				
	estimate	st. error	t value	p value
(Intercept)	**2.71*** **	0.11	25.02	<0.0001
Sem. density	**-0.35.**	0.16	-2.23	<0.1
Token freq.	-0.13	0.16	-0.89	0.42

Table 5: Fixed effects table for the first two models.

(Coer - imp) ∼sem. dens + type freq.				
	estimate	st. error	t value	p value
(Intercept)	**1.69*** **	0.15	10.87	<0.0001
Sem. density	**0.86***	0.22	3.38	<0.01
Type freq.	**0.47.**	0.22	2.1	<0.1
(Coer – imp) ∼sem. dens. + tok. freq.				
	estimate	st. error	t value	p value
(Intercept)	**1.69*** **	0.14	33.33	<0.0001
Sem. density	**0.91***	0.2	4.59	<0.001
Token freq.	**0.54***	0.2	2.71	<0.01

Table 6: Fixed effects table for the second two models.

tion between semantically incompatible and partially compatible formulations, whereas higher coercibility is only affected by semantic density.

We interpret this finding in light of the *upward strengthening hypothesis* (Hilpert, 2015), according to which a novel occurrence of a linguistic unit strengthens a superior node (i.e., the abstract Cxn) only if the former is categorized 'as an instance of a more abstract Cxn. If this categorization is not performed, or only superficially so, no upward strengthening will take place' (Hilpert, 2015, p.38). Higher coercibility is hence not affected by frequency of the Cxn because of the 'intermediate' grammaticality level of coercion, which does not allow unambiguous categorization. Coercion sentences result more natural if the target Cxn is observed with verbs belonging to similar semantic classes or subclasses, which increases Cxn semantic density. We could therefore assume that coercion effects in Italian elicit a *partial categorization*. The effect of semantic density, however, only explains part of the data. In fact, visual inspection of the relation between semantic density and the estimates of table 3 reveals that this effect does not explain the high coercibility of IM, or the low values of CO Cxns (see Figure 5).

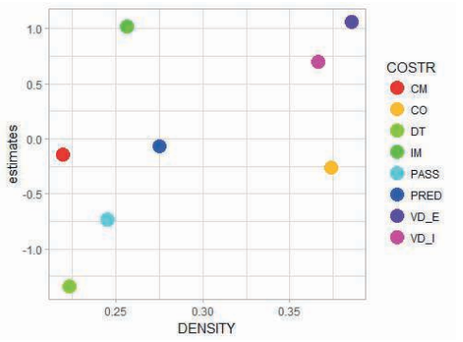


Figure 5: relation semantic density- estimates

All things considered, semantic properties (modelled with distributional vectors) of Cxns (e.g., its density) are only one of the factors influencing speakers processing and recognition of coercion effects. In fact, it has been argued that Romance languages are more *valency driven* than English (and Germanic languages in general) (Perek and Hilpert, 2014). The results of both experiments provide substantial evidence for an integrated account of Italian coercion effects, which should consider not only the properties of the general abstract Cxn, but rather the interaction of the mismatching verb with Cxn meaning.

These result also have interesting implications to understand the cognitive mechanisms underlying Cxn flexibility and productivity. In fact, these findings support the idea that Cxn meaning is abstracted from the semantics of prototypically occurring verbs. As we saw, several studies have argued in favour of this hypothesis for English, but the fact that we were able to adapt it to Italian suggests that the factors driving the acquisition of Cxns are - at least partially - not language-specific but rather general cognitive processes.

Acknowledgments:

The authors thank Lucia Passaro and Florent Perek for their help and valuable suggestions.

References

Libby Barak and Adele E. Goldberg. 2017. Modeling the Partial Productivity of Constructions.

Jóhanna Barðdal. 2008. *Productivity: Evidence from Case and Argument Structure in Icelandic*. 12.

Jóhanna Barddal. 2006. Predicting the Productivity of Argument Structure Constructions. *Annual Meeting of the Berkeley Linguistics Society*, 32(1):467, October.

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

Kamil Bartoń, 2013. *MuMIn: multi-model inference, R package version 1.9.13*. CRAN http://CRAN.R-project.org/package=MuMIn.

Hans Christian Boas and Francisco Gonzálvez-García, editors. 2014. *Romance perspectives on construction grammar*. Number volume 15 in Constructional approaches to language. John Benjamins Publishing Company, Amsterdam ; Philadelphia.

Hans C. Boas. 2011. Coercion and leaking argument structures in Construction Grammar. *Linguistics*, 49(6), January.

J. Bybee. 2006. *Frequency of Use and the Organization of Language*. Oxford University Press.

Joan L Bybee. 2013. Usage-based theory and exemplar representations of constructions. In *The Oxford handbook of construction grammar*.

Devin Casenhiser and Adele E Goldberg. 2005. Fast mapping between a phrasal form and meaning. *Developmental Science*, 8(6):500–508.

Michela Cennamo and Claudia Fabrizio, 2013. *Italian Valency Patterns*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Penelope Eckert. 2017. Age as a Sociolinguistic Variable. In *The Handbook of Sociolinguistics*, pages 151–167. Wiley-Blackwell.

Adele E. Goldberg. 1999. The emergence of the semantics of argument structure constructions. In *The emergence of language*, pages 215–230. Psychology Press.

Adele E. Goldberg. 2006. *Constructions at work: the nature of generalization in language*. Oxford linguistics. Oxford University Press, Oxford ; New York.

Martin Hilpert. 2015. From hand-carved to computer-based: Noun-participle compounding and the upward strengthening hypothesis. *Cognitive Linguistics*, 26(1), January.

Suzanne Kemmer and Soyeon Yoon. 2013. Rethinking coercion as a cognitive phenomenon: Data from processing, frequency, and acceptability judgments.

Suzanne Kemmer. 2008. September. new dimensions of dimensions: Frequency, productivity, domains and coercion. In *meeting of Cognitive Linguistics Between Universality and Variation, Dubrovnik, Croatia*.

Alexandra Kuznetsova, Per B. Brockhoff, and Rune H. B. Christensen. 2017. lmerTest package: Tests in linear mixed effects models. *Journal of Statistical Software*, 82(13):1–26.

W. Labov. 2001. *Principles of Linguistic Change, Social Factors*. Principles of Linguistic Change. Wiley.

Peter Lauwers and Dominique Willems. 2011. Coercion: Definition and challenges, current approaches, and new trends. *Linguistics*, 49(6), January.

Gianluca E. Lebani and Alessandro Lenci. 2017. Modelling the Meaning of Argument Constructions with Distributional Semantics. In *Proceedings of the AAAI 2017 Spring Symposium on Computational Construction Grammar and Natural Language Understanding*, pages 197–204.

Alessandro Lenci, Gabriella Lapesa, and Giulia Bonansinga. 2012. Lexit: A computational resource on italian argument structure. In *LREC*.

Alessandro Lenci. 2018. Distributional Models of Word Meaning. *Annual Review of Linguistics*, 4:151–171.

Laura A. Michaelis. 2004. Type shifting in construction grammar: An integrated approach to aspectual coercion. *Cognitive Linguistics*, 15(1):1–67, January.

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.

Florent Perek and Adele E. Goldberg. 2017. Linguistic generalization on the basis of function and constraints on the basis of statistical preemption. *Cognition*, 168:276–293.

Florent Perek and Martin Hilpert. 2014. Constructional tolerance: Cross-linguistic differences in the acceptability of non-conventional uses of constructions. *Constructions and Frames*, 6(2):266–304.

Florent Perek. 2016. Using distributional semantics to study syntactic productivity in diachrony: A case study. *Linguistics*, 54(1):149–188.

Giulia Rambelli, Alessandro Lenci, and Thierry Poibeau. 2017. UDLex: Towards Cross-language Subcategorization Lexicons. In *Proceedings of*

the Fourth International Conference on Dependency Linguistics (Depling 2017), September 18-20, 2017, Università di Pisa, Italy, pages 207–217. Linköping University Electronic Press.

Henrik Singmann, Ben Bolker, Jake Westfall, and Frederik Aust, 2016. *afex: Analysis of Factorial Experiments*. R package version 0.16-1.

Suzanne Evans Wagner. 2012. Age Grading in Sociolinguistic Theory: Age Grading in Sociolinguistic Theory. *Language and Linguistics Compass*, 6(6):371–382, June.

Soyeon Yoon. 2016. Gradable nature of semantic compatibility and coercion: A usage-based approach. *Linguistic Research*, 33(1):95–134, March.

Arne Zeschel. 2012. *Incipient productivity: a construction-based approach to linguistic creativity*. Number 49 in Cognitive linguistics research. De Gruyter Mouton, Berlin ; Boston.

The SEEMPAD Dataset for Emphatic and Persuasive Argumentation

Elena Cabrio and **Serena Villata**
Université Côte d'Azur, Inria, CNRS, I3S, France
elena.cabrio@unice.fr ; villata@i3s.unice.fr

Abstract

English. Emotions play an important role in argumentation as humans mix rational and emotional attitudes when they argue with each other to take decisions. The SEEMPAD project aims at investigating the role of emotions in human argumentation. In this paper, we present a resource resulting from two field experiments involving humans in debates, where arguments put forward during such debates are annotated with the emotions felt by the participants. In addition, in the second experiment, one of the debaters plays the role of the *persuader*, to convince the other participants about the goodness of her viewpoint applying different persuasion strategies. To the best of our knowledge, this is the first dataset of arguments annotated with the emotions collected from the participants of a debate, using facial emotion recognition tools.

Italiano. *Le emozioni giocano un ruolo importante nell'argomentazione in quanto gli esseri umani uniscono atteggiamenti razionali ad atteggiamenti puramente emotivi quando discutono tra loro per prendere decisioni. Il progetto SEEMPAD si propone di studiare il ruolo delle emozioni nell'argomentazione umana. In questo articolo, presentiamo una risorsa ottenuta tramite due esperimenti empirici che coinvolgono le persone nei dibattiti. Gli argomenti presentati durante tali dibattiti sono annotati con le emozioni provate dai partecipanti nel momento in cui l'argomento viene proposto nella discussione. Inoltre, durante il secondo esperimento, uno dei partecipanti svolge il ruolo di persuasore, al fine di convincere gli altri partecipanti della bontá del suo punto di vista applicando diverse strategie di persuasione. Questa risorsa è peculiare nel suo genere, ed è l'unica a contenere argomenti annotati con le emozioni provate dai partecipanti durante un dibattito (emozioni registrate tramite strumenti di riconoscimento automatico delle emozioni facciali).*

1 Introduction

Argumentation in Artificial Intelligence (AI) is defined as a formal framework to support decision making (Rahwan and Simari, 2009; Atkinson et al., 2017). In this context, argumentation is used to achieve the so called *critical thinking*. However, humans are proved to behave differently as they mix rational and *emotional* attitudes.

In order to study the role emotions play in argumentation, we proposed an empirical evaluation of the connection between argumentation and emotions in online debate interactions (Villata et al., 2017; Villata et al., 2018). In particular, in the context of the SEEMPAD project,[1] we designed a field experiment (Villata et al., 2017) with human participants which investigates the correspondences between the arguments and their relations (i.e., support and attack) put forward during a debate, and the emotions detected by facial emotion recognition systems in the debaters. In addition, given the importance of persuasion in argumentation, we also designed a second field experiment (Villata et al., 2018) to study the correlation between the arguments, the relations between them, the emotions detected on the participants, and one of the classical persuasion strategies proposed by Aristotle in rethoric (i.e., *logos*, *ethos*, and *pathos*), played by some participants in the debate to convince the others. In our studies, we selected a behavioral method to extract

[1] https://project.inria.fr/seempad/

the emotional manifestations. We used a set of webcams (one for each participant in the discussion) whose recordings have been analyzed with the FaceReader software[2] to detect a set of discrete emotions from facial expressions (i.e., happiness, anger, fear, sadness, disgust, and surprise). Participants were placed far from each other, and they were debating through a purely text-based online debate chat (IRC). As a post-processing phase, we aligned the textual arguments the debaters proposed in the chat with the emotions the debaters were feeling while these arguments have been proposed in the debate.

In this paper, we describe the two annotated resources resulting from this post-processing of the data we collected in our two field experiments. Our resource, called the SEEMPAD resource, is composed of two different annotated datasets, one for each of these experiments[3]. The datasets collect all the arguments put forward during the debates. These arguments have been paired by *attack* and *support* relations, as in standard Argument Mining annotations (Cabrio and Villata, 2018; Lippi and Torroni, 2016). Moreover, arguments are annotated with the source of the argument, and the emotional status captured from all the participants, when the arguments are put forward in the debate.

To the best of our knowledge, this is the first argumentation dataset annotated with the emotions captured from the output of facial emotion recognition tools. In addition, this resource can be used both for argument mining tasks (i.e., relation prediction), and for emotion classification in text, where instances of text annotated with the emotions detected on the participants are usually not available. Finally, text-based emotion classification would benefit from the different annotation layers that are present in our dataset.

In the reminder of the paper, Sections 2 and 3 describe the dataset resulting from the two field experiments. Conclusions end the paper.

2 Dataset of argument pairs associated with the speaker's emotional status

This section describes the dataset of textual arguments we have created from the debates among the participants in Experiment 1 (Villata et al., 2017). The dataset is composed of four main layers: *(i)* the basic annotation of the arguments[4] proposed in each debate (i.e., the annotation in xml of the debate flow downloaded from the debate platform); *(ii)* the annotation of the relations of support and attack among the arguments; *(iii)* starting from the basic annotation of the arguments, the annotation of each argument with the emotions felt by each participant involved in the debate; and *(iv)* starting from the basic annotation, the opinion of each participant about the debated topic at the beginning, in the middle and at the end of debate is extracted and annotated with its polarity.

The *basic* dataset is composed of 598 different arguments proposed by the participants in 12 different debates. The debated issues and the number of arguments for each debate are reported in Table 1. We selected the topics of the debates among the set of popular discussions addressed in online debate platforms like iDebate[5] and DebateGraph[6].

In the dataset, each argument proposed in the debate is annotated with an id, the participant putting this argument on the table, and the time interval the argument has been proposed.[7] Then, arguments pairs have been annotated with the relation holding between them, i.e., support or attack. For each debate we apply the following procedure, validated in (Cabrio and Villata, 2013):

1. the main issue (i.e., the issue of the debate proposed by the moderator) is considered as the starting argument;
2. each opinion is extracted and considered as an argument;
3. since *attack* and *support* are binary relations, the arguments are coupled with:
   a the starting argument, or
   b other arguments in the same discussion to which the most recent argument refers

[2] https://www.noldus.com/human-behavior-research/products/facereader

[3] Available at http://project.inria.fr/seempad/datasets/

[4] Note that we annotated as an argument each utterance proposed by the participants in the debate. We did not need then to define guidelines to distinguish arguments or their components in the debate, as it is usually done in the Argument Mining field (Cabrio and Villata, 2018).

[5] http://idebate.org/

[6] www.debategraph.org/

[7] Note that when the argument was put forward by the debater in one single utterance the two time instances (i.e., time-from and time-to) coincide. We used the time interval only when the argument was composed of several separated utterances put forward in the chat across some minutes.

(e.g., when an argument proposed by a certain user supports or attacks an argument previously expressed by another user);

4. the resulting pairs of arguments are then tagged with the appropriate relation, i.e., *attack* or *support*.

To show a step-by-step application of the procedure, let us consider the debated issue *Ban Animal Testing*. At step 1, we consider the issue of the debate proposed by the moderator as the starting argument (a):

(a) *The topic of the first debate is that animal testing should be banned.*

Then, at step 2, we extract all the users opinions concerning this issue (both pro and con), e.g., (b), (c) and (d):

(b) *I don't think the animal testing should be banned, but researchers should reduce the pain to the animal.*

(c) *I totally agree with that.*

(d) *I think that using animals for different kind of experience is the only way to test the accuracy of the method or drugs. I cannot see any difference between using animals for this kind of purpose and eating their meat.*

(e) *Animals are not able to express the result of the medical treatment but humans can.*

At step 3a we couple the arguments (b) and (d) with the starting issue since they are directly linked with it, and at step 3b we couple argument (c) with argument (b), and argument (e) with argument (d) since they follow one another in the discussion. At step 4, the resulting pairs of arguments are then tagged by one annotator with the appropriate relation, i.e.: **(b)** *attacks* **(a)**, **(d)** *attacks* **(a)**, **(c)** *supports* **(b)** and **(e)** *attacks* **(d)**. The reader may argue about the existence of a relation (i.e., a support) between (c) and (d), where (d) supports (c). However, in this case, no relation holds as argument (d) does not really supports argument (c), which basically share the same semantic content of argument (b). Therefore, as no relation holds between (b) and (d), no relation holds either between (c) and (d). We decided to not annotate the supports/attacks between arguments proposed by the same participant (e.g., situations where participants are contradicting themselves). Note that this does not mean that we assume that such situations do not arise: no restriction was imposed to the participants of the debates, so situations where a participant attacked/supported her own arguments are represented in our dataset. The same annotation task has been independently carried out also by a second annotator on a sample of 100 pairs (randomly extracted), obtaining an IAA of $\kappa = 0.82$. The IAA is computed on the assignement of the label "support" or "attack" to the same set of pairs provided to the two annotators.

Topic	#arg	#pair	#att	#sup
BAN ANIMAL TESTING	49	28	18	10
GO NUCLEAR	40	24	15	9
HOUSEWIVES SHOULD BE PAID	42	18	11	7
RELIGION DOES MORE HARM THAN GOOD	46	23	11	12
ADVERTISING IS HARMFUL	71	16	6	10
BULLIES ARE LEGALLY RESPONSIBLE	71	12	3	9
DISTRIBUTE CONDOMS IN SCHOOLS	68	27	11	16
ENCOURAGE FEWER PEOPLE TO GO TO THE UNIVERSITY	55	14	7	7
FEAR GOVERNMENT POWER OVER INTERNET	41	32	18	14
BAN PARTIAL BIRTH ABORTIONS	41	26	15	11
USE RACIAL PROFILING FOR AIRPORT SECURITY	31	10	1	9
CANNABIS SHOULD BE LEGALIZED	43	33	20	13
TOTAL	**598**	**263**	**136**	**127**

Table 1: Dataset of argument pairs and emotions (#arg: number of arguments, #pairs: number of pairs, #att: number of attacks, #supp: number of supports).

Table 1 reports on the number of arguments and pairs we extracted applying the methodology described before. In total, our dataset contains 598 different arguments and 263 argument pairs (127 expressing the *support* relation and 136 the *attack* relation among the involved arguments).

The dataset resulting from such annotation adds to all previously annotated information (i.e., argument id, the argument's owner, argument's relations with the other arguments (attack, support)), the dominant emotion detected using the FaceReader system for each participant in the debate. We investigate the correlation between arguments and emotions in the debates, and a data analysis has been performed to determine the proportions of emotions for all participants. For more details about the correlation between emotions and arguments, we refer the interested reader to (Villata et

al., 2017).

An example, from the debate about the topic "Religion does more harm than good" where arguments are annotated with emotions, is as follows:

```
<argument id="30" debate_id="4" partici-
pant="4" time-from="20:43" time-to="20:
43" emotion_p1="neutral" emotion_p2=
"neutral" emotion_p3="neutral" emotion_
p4="neutral"> Indeed but there exist
some advocates of the devil like Bernard
Levi who is decomposing arabic countries.
</argument>

<argument id="31" debate_id="4" partici-
pant="1" time-from="20:43" time-to="20:
43" emotion_p1="angry" emotion_p2="neu-
tral" emotion_p3="angry" emotion_p4=
"disgusted">I don't totally agree with
you Participant2: science and religion
don't explain each other, they tend to
explain the world but in two different
ways.</argument>
```

In this example, the argument "I don't totally agree with you Participant2: science and religion don't explain each other, they tend to explain the world but in two different ways." is proposed by Participant 4 in the debate, and the emotions resulting from this argument when it has been put forward in the chat are *neutrality* for Participant 2, *anger* for Participant 1 and Participant 3, and *disgust* for Participant 4.

Finally, as an additional annotation layer, for each participant we have selected one argument at the beginning of the debate, one argument in the middle of the discussion, and one argument at the end of the debate. These arguments are then annotated by the annotators with their *sentiment classification* with respect to the issue of the debate: *negative*, *positive*, or *undecided*. The negative sentiment is assigned to an argument when the opinion expressed in such argument is against the debated topic, while the positive sentiment label is assigned when the argument expresses a viewpoint that is in favor of the debated issue. The undecided sentiment is assigned when the argument does not express a precise opinion in favor or against the debated topic. Selected arguments are evaluated as the most representative arguments proposed by each participant to convey her own opinion, in the three distinct moments of the debate. The rationale is that this annotation allows to easily detect when a participant has changed her mind with respect to the debated topic. An example is provided below, where Participant4 starts the debate being undecided and then turns to be positive about banning partial birth abortions in the middle and at the end of the debate:

```
<arg id="5" participant="4" time-from=
"20:36" time-to="20:36" polarity="undeci-
ded">Description's gruesome but does the
fetus fully lives at that point and the-
refore, conscious of something ? Hard to
answer. If yes, I might have an hesita-
tion to  accept it. If not, the woman is
probably mature enough to judge.
</argument>

<arg id="24" participant="4" time-from=
"20:46" time-to="20:46" polarity="positi-
ve">In the animal world, malformed or
sick babies are systematically abandoned.
</argument>

<arg id="38" participant="4" time-from=
"20:52" time-to="20:52" polarity="positi-
ve">Abortion is legal and it doesn't mat-
ter much when and how. It's an individual
choice for whatever reason it might be.
</argument>
```

3 Dataset of arguments biased by persuasive strategies

We now describe the corpus of textual arguments, about other discussion topics, collected during Experiment 2 (Villata et al., 2018), in which, together with the participants of the experiment, a *persuader* (PP) was involved to convince the other participants about the goodness of her viewpoint, applying different persuasion strategies. Three kinds of argumentative persuasion exist since Aristotle: *Ethos*, *Logos*, and *Pathos* (Ross and Roberts, 2010; Walton, 2007; Allwood, 2016). Ethos deals with the character of the speaker, whose intent is to appear credible. The main influencing factors for Ethos encompass elements such as vocabulary, and social aspects like rank or popularity. Additionally, the speaker can use statements to position himself and to reveal social hierarchies. Logos is the appeal to logical reason: the speaker wants to present an argument that appears to be sound to the audience. For the argumentation, the focus of interest is on the arguments, the argument schemes, the different forms of proof and the reasoning. Pathos encompasses the emotional influence on the audience. If the goal of argumentation is to persuade the audience, then it is necessary to put the audience in the appropriate emotional states. The public speaker has several possibilities to awaken emotions in the audience, like techniques and presentation styles (e.g., storytelling), reducing the ability of the audience to

Dataset						
Topic	**Strategy**	**PP position**	**#arg**	**#pair**	**#att**	**#sup**
SINGLE SEX-SCHOOLS ARE GOOD FOR EDUCATION	Logos	Pro	62	20	12	8
SALE OF HUMAN ORGANS SHOULD BE LEGALIZED	Pathos	Con	37	6	1	5
PARENTS ARE ACCOUNTABLE FOR REFUSING TO VACCINATE THEIR CHILDREN	Logos	Pro	74	17	6	11
THERE SHOULD BE GUN RIGHTS	Ethos	Con	94	24	12	12
GO NUCLEAR	Logos	Pro	87	9	8	1
RELIGION DOES MORE HARM THAN GOOD	Pathos	Con	59	14	6	8
ASSISTED SUICIDE SHOULD BE LEGALIZED	Ethos	Pro	102	29	20	9
USE RACIAL PROFILING - AIRPORT	Logos	Con	34	3	0	3
DEATH PENALTY SHOULD BE SUPPORTED	Pathos	Con	128	27	7	20
TORTURE SHOULD BE USED ON TERRORISTS	Logos	Pro	114	13	2	11
TOTAL			**791**	**162**	**74**	**88**

Table 2: Dataset of argument pairs and persuasion strategies (PP position: stance of the persuader with respect to the topic of the debate).

be critical or to reason.[8] It is worth noticing that the persuasive strategies are not always mutually exclusive in real world scenario, however, for the sake of simplicity, we consider in this paper that when one of the strategies is applied the other do not hold. In addition to a persuasion strategy, the persuader participated into the debate with a precise stance (pro or con) with respect to the debated issue. Such stance does not change during the debate.

Each argument is annotated with the following elements: debate identifier, argument identifier, participant, and time in which it has been published. For each debate, pairs have been created following the same methodology described in Section 2, and all the relations of attack and support between the arguments proposed by the persuader and those of the other participants are annotated. In this way, we are able to investigate the reactions to PP strategy by tracking the proposed arguments in the debate and the mental engagement index of the other participants. An example of Ethos strategy used against gun rights is the following:

```
<arg id="16" debate_id="8" participant="5"
time="19:46:41"> I've been working in the
educational field in USA, and there no-
thing worse than a kid talking about the
gun of his father. As you cannot say "the
right to carry guns is for people without
a kid only". Then no right at all.
</argument>
```

Table 2 describes this second dataset. Ten topics of debate were selected from highly debated ones in the mentioned online debate platforms, to avoid proposing topics of no interest for the participants. In total, 791 arguments, and 162 arguments pairs (74 linked by an attack relation and 88 by a support one) were collected and annotated. The number of proposed arguments varies a lot depending on the participants (some were more active, others proposed very few arguments even if solicited), as well as the number of attacks/supports between the arguments. We computed the IAA for the relation annotation task on 1/3 of the pairs of the dataset (54 randomly extracted pairs), obtaining $\kappa = 0.83$.

[8] For more details, refer to the work of K. Budzynska.

4 Conclusions

This paper presented the SEEMPAD resource for empathic and persuasive argumentation. These datasets have been built on the data resulting from two field experiments on humans to assess the impact of emotions during the argumentation in online debates. Several Natural Language Processing tasks can be can be thought on this dataset. First of all, given that the dataset resulting from the Experiment 1 is a gold standard of arguments annotated with emotions, systems for emotion classification can use it as a benchmark for evaluation. In addition, a comparison of systems' performances on this data compared with the standard dataset for emotion classification would be interesting, given that in SEEMPAD emotions have not been manually annotated but they have been captured from the participants' facial emotion expressions. Second, the dataset from Experiment 2 can be used to address a new task in argument mining, namely persuasive strategy detection, in line with the work of (Duthie and Budzynska, 2018) and (Habernal and Gurevych, 2016).

References

Jens Allwood. 2016. Argumentation, activity and culture. In *Proceedings of COMMA 2016*, page 3.

Katie Atkinson, Pietro Baroni, Massimiliano Giacomin, Anthony Hunter, Henry Prakken, Chris Reed, Guillermo Simari, Matthias Thimm, and Serena Villata. 2017. Towards artificial argumentation. *AI Magazine*, 38(3):25–36.

Elena Cabrio and Serena Villata. 2013. A natural language bipolar argumentation approach to support users in online debate interactions. *Argument & Computation*, 4(3):209–230.

Elena Cabrio and Serena Villata. 2018. Five years of argument mining: a data-driven analysis. In *Proc. of IJCAI 2018*, pages 5427–5433.

Rory Duthie and Katarzyna Budzynska. 2018. A deep modular RNN approach for ethos mining. In *Proc. of IJCAI 2018*, pages 4041–4047.

Ivan Habernal and Iryna Gurevych. 2016. Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional LSTM. In *Proc. of ACL 2016*.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Techn.*, 16(2):10:1–10:25.

Iyad Rahwan and Guillermo R. Simari. 2009. *Argumentation in Artificial Intelligence*. Springer.

W.D. Ross and W.R. Roberts. 2010. *Rhetoric - Aristotle*. Cosimo Classics Philosophy.

Serena Villata, Elena Cabrio, Imène Jraidi, Sahbi Benlamine, Maher Chaouachi, Claude Frasson, and Fabien Gandon. 2017. Emotions and personality traits in argumentation: An empirical evaluation. *Argument & Computation*, 8(1):61–87.

Serena Villata, Sahbi Benlamine, Elena Cabrio, Claude Frasson, and Fabien Gandon. 2018. Assessing persuasion in argumentation through emotions and mental states. In *Proc. of FLAIRS 2018*, pages 134–139.

Douglas N. Walton. 2007. *Media argumentation - dialect, persuasion and rhetoric*. Cambridge University Press.

Italian Event Detection Goes Deep Learning

Tommaso Caselli
CLCG, Rijksuniversiteit Groningen Oude Kijk in't Jaatsraaat, 26
9712 EK Groningen (NL)
t.caselli@{rug.nl}{gmail.com}

Abstract

English. This paper reports on a set of experiments with different word embeddings to initialize a state-of-the-art Bi-LSTM-CRF network for event detection and classification in Italian, following the EVENTI evaluation exercise. The network obtains a new state-of-the-art result by improving the F1 score for detection of 1.3 points, and of 6.5 points for classification, by using a single step approach. The results also provide further evidence that embeddings have a major impact on the performance of such architectures.

Italiano. *Questo contributo descrive una serie di esperimenti con diverse rappresentazioni distribuzionali di parole (word embeddings) per inizializzare una rete neurale stato dell'arte di tipo Bi-LSTM-CRF per il riconoscimento e la classificazione di eventi in italiano, in base all'esercizio di valutazione EVENTI. La rete migliora lo stato dell'arte di 1.3 punti di F1 per il riconoscimento, e di 6.5 punti per la classificazione, affrontando il compito in un unico sistema. L'analisi dei risultati fornisce ulteriore supporto al fatto che le rappresentazioni distribuzionali di parole hanno un impatto molto alto nei risultati di queste architetture.*

1 Introduction

Current societies are exposed to a continuous flow of information that results in a large production of data (e.g. news articles, micro-blogs, social media posts, among others), at different moments in time. In addition to this, the consumption of information has dramatically changed: more and more people directly access information through social media platforms (e.g. Facebook and Twitter), and are less and less exposed to a diversity of perspectives and opinions. The combination of these factors may easily result in *information overload* and impenetrable "*filter bubbles*". Events, i.e. things that happen or hold as true in the world, are the basic components of such data stream. Being able to correctly identify and classify them plays a major role to develop robust solutions to deal with the current stream of data (e.g. the storyline framework (Vossen et al., 2015)), as well to improve the performance of many Natural Language Processing (NLP) applications such as automatic summarization and question answering (Q.A.).

Event detection and classification has seen a growing interest in the NLP community thanks to the availability of annotated corpora (LDC, 2005; Pustejovsky et al., 2003a; O'Gorman et al., 2016; Cybulska and Vossen, 2014) and evaluation campaigns (Verhagen et al., 2007; Verhagen et al., 2010; UzZaman et al., 2013; Bethard et al., 2015; Bethard et al., 2016; Minard et al., 2015). In the context of the 2014 EVALITA Workshop, the EVENTI evaluation exercise (Caselli et al., 2014)[1] was organized to promote research in Italian Temporal Processing, of which event detection and classification is a core subtask.

Since the EVENTI campaign, there has been a lack of further research, especially in the application of deep learning models to this task in Italian. The contributions of this paper are the followings: i.) the adaptation of a state-of-the-art sequence to sequence (seq2seq) neural system to event detection and classification for Italian in a single step approach; ii.) an investigation on the quality of existing Italian word embeddings for this task; iii.) a comparison against a state-of-the-art discrete classifier. The pre-trained models and scripts running

[1] https://sites.google.com/site/eventievalita2014/

the system (or re-train it) are publicly available. [2].

2 Task Description

We follow the formulation of the task as specified in the EVENTI exercise: determine the extent and the class of event mentions in a text, according to the It-TimeML <EVENT> tag definition (Subtask B in EVENTI).

In EVENTI, the tag <EVENT> is applied to every linguistic expression denoting a situation that happens or occurs, or a state in which something obtains or holds true, regardless of the specific parts-of-speech that may realize it. EVENTI distinguishes between single token and multi-tokens events, where the latter are restricted to specific cases of eventive multi-word expressions in lexicographic dictionaries (e.g. "*fare le valigie*" [to pack]), verbal periphrases (e.g. "*(essere) in grado di*" [(to be) able to]; "*c'è*" [there is]), and named events (e.g. "*la strage di Beslan*" [Beslan school siege]).

Each event is further assigned to one of 7 possible classes, namely: OCCURRENCE, ASPECTUAL, PERCEPTION, REPORTING, I(NTESIONAL)_STATE, I(NTENSIONAL)_ACTION, and STATE. These classes are derived from the English TimeML Annotation Guidelines (Pustejovsky et al., 2003). The TimeML event classes distinguishes with respect to other classifications, such as ACE (LDC, 2005) or FrameNet (Baker et al., 1998), because they expresses relationships the target event participates in (such as factual, evidential, reported, intensional) rather than semantic categories denoting the meaning of the event. This means that the EVENT classes are assigned by taking into account both the semantic and the syntactic context of occurrence of the target event. Readers are referred to the EVENTI Annotation Guidelines for more details[3].

2.1 Dataset

The EVENTI corpus consists of three datasets: the Main Task training data, the Main task test data, and the Pilot task test data. The Main Task data are on contemporary news articles, while the Pilot Task on historical news articles. For our experiments, we focused only on the Main Task. In addition to the training and test data, we have created also a Main Task development set by excluding from the training data all the articles that composed the test data of the Italian dataset at the SemEval 2010 TempEval-2 campaign (Verhagen et al., 2010). The new partition of the corpus results in the following distribution of the <EVENT> tag: i) 17,528 events in the training data, of which 1,207 are multi-token mentions; ii.) 301 events in the development set, of which 13 are multi-token mentions; and finally, iii.) 3,798 events in the Main task test, of which 271 are multi-token mentions.

Tables 1 and 2 report, respectively, the distribution of the events per token part-of speech (POS) and per event class. Not surprisingly, verbs are the largest annotated category, followed by nouns, adjectives, and prepositional phrases. Such a distribution reflects both a kind of "natural" distribution of the realization of events in an Indo-european language, and, at the same time, specific annotation choices. For instance, adjectives have been annotated only when in a predicative position and when introduced by a copula or a copular construction. As for the classes, OCCURRENCE and STATE represent the large majority of all events, followed by the intensional ones (I_STATE and I_ACTION), expressing some factual relationship between the target events and their arguments, and finally the others (REPORTING, ASPECTUAL, and PERCEPTION).

3 System and Experiments

We adapted a publicly available Bi-LSTM network with a CRF classifier as last layer (Reimers and Gurevych, 2017). [4] (Reimers and Gurevych, 2017) demonstrated that word embeddings, among other hyper-parameters, have a major impact on the performance of the network, regardless of the specific task. On the basis of these experimental observations, we decided to investigate the impact of different Italian word embeddings for the Subtask B Main Task of the EVENTI exercise. We thus selected 5 word embeddings for Italian to initialize the network, differentiating one with respect to each other either for the representation model used (`word2vec` *vs.* GloVe; CBOW *vs.* skip-gram), dimensionality (300 *vs.* 100), or corpora used for their generation (Italian

[2] https://github.com/tommasoc80/Event_detection_CLiC-it2018

[3] https://sites.google.com/site/eventievalita2014/file-cabinet

[4] https://github.com/UKPLab/emnlp2017-bilstm-cnn-crf

POS	Training	Dev.	Test
Noun	6,710	111	1,499
Verb	11,269	193	2,426
Adjective	610	9	118
Preposition	146	1	25
Overall Event Tokens	18,735	314	4,068

Table 1: Distribution of the event mentions per POS per token in all datasets of the EVENTI corpus.

Class	Training	Dev.	Test
OCCURRENCE	9,041	162	1,949
ASPECTUAL	446	14	107
I_STATE	1,599	29	355
I_ACTION	1,476	25	357
PERCEPTION	162	2	37
REPORTING	714	8	149
STATE	4,090	61	843
Overall Events	17,528	301	3,798

Table 2: Distribution of the event mentions per class in all datasets of the EVENTI corpus.

Wikipedia *vs*. crawled web document *vs*. large textual corpora or archives):

- Berardi2015_w2v (Berardi et al., 2015): 300 dimension word embeddings generated using the `word2vec` (Mikolov et al., 2013) skip-gram model [5] from the Italian Wikipedia;
- Berardi2015_glove (Berardi et al., 2015): 300 dimensions word embeddings generated using the GloVe model (Pennington et al., 2014) from the Italian Wikipedia[6];
- Fastext-It: 300 dimension word embeddings from the Italian Wikipedia [7] obtained using Bojanovsky's skip-gram model representation (Bojanowski et al., 2016), where each word is represented as a bag of character n-grams [8];
- ILC-ItWack (Cimino and Dell'Orletta, 2016): 300 dimension word embeddings generated by using the `word2vec` CBOW model [9] from the ItWack corpus;
- DH-FBK_100 (Tonelli et al., 2017): 100 dimension word and phrase embeddings, generated using the `word2vec` and `phrase2vec` models, from 1.3 billion word corpus (Italian Wikipedia, OpenSubtitles2016 (Lison and Tiedemann, 2016), PAISA corpus [10], and the Gazzetta Ufficiale).

As for the other parameters, the network maintains the optimized configurations used for the event detection task for English (Reimers and Gurevych, 2017): two LSTM layers of 100 units each, *Nadam* optimizer, variational dropout (0.5, 0.5), with gradient normalization (τ = 1), and batch size of 8. Character-level embeddings, learned using a Convolutional Neural Network (CNN) (Ma and Hovy, 2016), are concatenated with the word embedding vector to feed into the LSTM network. Final layer of the network is a CRF classifier.

Evaluation is conducted using the EVENTI evaluation framework. Standard Precision, Recall, and F1 apply for the event detection. Given that the extent of an event tag may be composed by more than one tokens, systems are evaluated both for strict match, i.e. one point only if all tokens which compose an <EVENT> tag are correctly identified, and relaxed match, i.e. one point for any correct overlap between the system output and the reference gold data. The classification aspect is evaluated using the F1-attribute score (UzZaman et al., 2013), that captures how well a system identify both the entity (extent) and attribute (i.e. class) together.

We approached the task in a single-step by detecting and classifying event mentions at once rather than in the standard two step approach, i.e. detection first and classification on top of the detected elements. The task is formulated as a seq2seq problem, by converting the original annotation format into an BIO scheme (Beginning, Inside, Outside), with the resulting alphabet being B-*class_label*, I-*class_label* and O. Example 1 below illustrates a simplified version of the problem for a short sentence:

(1)

input	problem	solution
Marco	(B-STATE | I-STATE | ... | O)	O
pensa	(B-STATE | I-STATE | ... | O)	B-ISTATE
di	(B-STATE | I-STATE | ... | O)	O
andare	(B-STATE | I-STATE | ... | O)	B-OCCUR
a	(B-STATE | I-STATE | ... | O)	O
casa	(B-STATE | I-STATE | ... | O)	O

[5] Parameters: negative sampling 10, context window 10

[6] Berardi2015_w2v and Berardi2015_glove uses a 2015 dump of the Italian Wikipedia

[7] Wikipedia dump not specified.

[8] `https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md`

[9] Parameters: context window 5.

[10] `http://www.corpusitaliano.it/`

	Strict Evaluation				Relaxed Evaluation			
Embedding Parameter	R	P	F1	F1-class	R	P	F1	F1-class
Berardi2015_w2v	0.868	0.868	0.868	0.705	0.892	0.892	0.892	0.725
Berardi2015_Glove	0.848	0.872	0.860	0.697	0.870	0.895	0.882	0.714
Fastext-It	**0.897**	0.863	**0.880**	**0.736**	**0.921**	0.887	**0.903**	**0.756**
ILC-ItWack	0.831	**0.884**	0.856	0.702	0.860	**0.914**	0.886	0.725
DH-FBK_100	0.855	0.859	0.857	0.685	0.881	0.885	0.883	0.705
FBK-HLT@EVENTI 2014	0.850	*0.884*	0.867	0.671	0.868	0.902	0.884	0.685

Table 3: Results for Bubtask B Main Task - Event detection and classification.

. (B-STATE | I-STATE | . . . | O) O

3.1 Results and Discussion

Results for the experiments are illustrated in Table 3. We also report the results of the best system that participated at EVENTI Subtask B, FBK-HLT (Mirza and Minard, 2014). FBK-HLT is a cascade of two SVM classifiers (one for detection and one for classification) based on rich linguistic features. Figure 1 plots charts comparing F1 scores of the network initialized with each of the five embeddings against the FBK-HLT system for the event detection and classification tasks, respectively.

The results of the Bi-LSTM-CRF network are varied in both evaluation configurations. The differences are mainly due to the embeddings used to initialize the network. The best embedding configuration is Fastext-It that differentiate from all the others for the approach used for generating the embeddings. Embedding's dimensionality impacts on the performances supporting the findings in (Reimers and Gurevych, 2017), but it seems that the quantity (and variety) of data used to generate the embeddings can have a mitigating effect, as shown by the results of the DH-FBK-100 configuration (especially in the classification subtask, and in the Recall scores for the event extent subtask). Coverage of the embeddings (and consequenlty, tokenization of the dataset and the embeddings) is a further aspect to keep into account, but it seems to have a minor impact with respect to dimensionality. It turns out that (Berardi et al., 2015)'s embeddings are those suffering the most from out of vocabulary (OVV) tokens (2.14% and 1.06% in training, 2.77% and 1.84% in test for the `word2vec` model and GloVe, respectively) with respect to the others. However, they still outperform DH-FBK_100 and ILC-ItWack, whose OVV are much lower (0.73% in training and 1.12% in test for DH-FBK_100; 0.74% in training and 0.83% in test for ILC-ItWack).

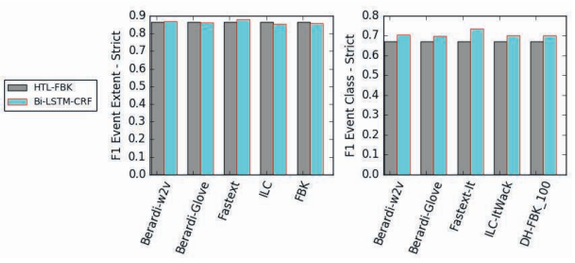


Figure 1: Plots of F1 scores of the Bi-LSTM-CRF systems against the FBK-HLT system for Event Extent (left side) and Event Class (right side). F1 scores refers to the

The network obtains the best F1 score, both for detection (F1 of 0.880 for strict evaluation and 0.903 for relaxed evaluation with Fastext-It embeddings) and for classification (F1-class of 0.756 for strict evaluation, and 0.751 for relaxed evaluation with Fastext-It embeddings). Although FBK-HLT suffers in the classification subtask, it qualifies as a highly competitive system for the detection subtask. By observing the strict F1 scores, FBK-HLT beats three configurations (DH-FBK-100, ILC-ItWack, Berardi2015_Glove) [11], almost equals one (Berardi2015_w2v) [12], and it is outperformed only by one (Fastext-It) [13]. In the relaxed evaluation setting, DH-FBK-100 is the only configuration that does not beat FBK-HLT (although the difference is only 0.001 point). Nevertheless, it is remarkable to observe that FBK-HLT has a very high Precision (0.902, relaxed evaluation mode), that is overcome by only one embedding configuration, ILC-ItWack. The results also indicates that word embeddings have a major contribution on Recall, supporting observations that distributed representations have better generalization capabilities than discrete feature vectors. This is further

[11] p-value < 0.005 only against Berardi2015_Glove and DH-FBK-100, with McNemar's test.

[12] p-value > 0.005 with McNemar's test.

[13] p-value < 0.005 with McNemar's test.

supported by the fact that these results are obtained using a single step approach, where the network has to deal with a total of 15 possible different labels.

We further compared the outputs of the best model, i.e. Fastext-It, against FBK-HLT. As for the event detection subtask, we have adopted an event-based analysis rather than a token based one, as this will provide better insights on errors concerning multi-token events and event parts-of-speech (see Table 1 for reference). [14] By analyzing the True Positives, we observe that the Fastext-It model has better performances than FBK-HLT with nouns (77.78% *vs*. 65.64%, respectively) and prepositional phrases (28.00% *vs*. 16.00%, respectively). Performances are very close for verbs (88.04% *vs*. 88.49%, respectively) and adjectives (80.50% *vs*. 79.66%, respectively). These results, especially those for prepositional phrases, indicates that the Bi-LSTM-CRF network structure and embeddings are also much more robust at detecting multi-tokens instances of events, and difficult realizations of events, such as nouns.

Concerning the classification, we focused on the mismatches between correctly identified events (extent layer) and class assignment. The Fastext-It model wrongly assigns the class to only 557 event tokens compared to the 729 cases for FBK-HLT. The distribution of the class errors, in terms of absolute numbers, is the same between the two systems, with the top three wrong classes being, in both cases, OCCURRENCE, I_ACTION and STATE. OCCURRENCE, not surprisingly, is the class that tends to be assigned more often by both systems, being also the most frequent. However, if FBK-HLT largely overgeneralizes OCCURRENCE (59.53% of all class errors), this corresponds to only one third of the errors (37.70%) in the Bi-LSTM-CRF network. Other notable differences concern I_ACTION (27.82% of errors for the Bi-LSTM-CRF *vs*. 17.28% for FBK-HLT), STATE (8.79% for the Bi-LSTM-CRF *vs*. 15.22% for FBK-HLT) and REPORTING (7.89% for the Bi-LSTM-CRF *vs*. 2.33% for FBK-HLT) classes.

4 Conclusion and Future Work

This paper has investigated the application of different word embeddings for the initialization of a state-of-the-art Bi-LSTM-CRF network to solve the event detection and classification task in Italian, according to the EVENTI exercise. We obtained new state-of-the-art results using the Fastext-It embeddings, and improved the F1-class score of 6.5 points in strict evaluation mode. As for the event detection subtask, we observe a limited improvement (+1.3 points in strict F1), mainly due to gains in Recall. Such results are extremely positive as the task has been modeled in a single step approach, i.e. detection and classification at once, for the first time in Italian. Further support that embeddings have a major impact in the performance of neural architectures is provided, as the variations in performance of the Bi-LSMT-CRF models show. This is due to a combination of factors such as dimensionality, (raw) data, and the method used for generating the embeddings.

Future work should focus on the development of embeddings that move away from the basic word level, integrating extra layers of linguistic analysis (e.g. syntactic dependencies) (Komninos and Manandhar, 2016), that have proven to be very powerful for the same task in English.

[14] Note that POS are manually tagged for events, not for their components.

Acknowledgments

The author wants to thank all researchers and research groups who made available their word embeddings and their code. Sharing is caring.

References

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

Giacomo Berardi, Andrea Esuli, and Diego Marcheggiani. 2015. Word embeddings go to italy: A comparison of models and training datasets. In *IIR*.

Steven Bethard, Leon Derczynski, Guergana Savova, James Pustejovsky, and Marc Verhagen. 2015. Semeval-2015 task 6: Clinical tempeval. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 806–814.

Steven Bethard, Guergana Savova, Wei-Te Chen, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2016. Semeval-2016 task 12: Clinical tempeval. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1052–1062.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vec-

tors with subword information. *arXiv preprint arXiv:1607.04606*.

T. Caselli, R. Sprugnoli, M. Speranza, and M. Monachini. 2014. Eventi.EValuation of Events and Temporal INformation at Evalita 2014. In C. Bosco, F. DellOrletta, S. Montemagni, and M. Simi, editors, *Evaluation of Natural Language and Speech Tools for Italian*, volume 1, pages 27–34. Pisa University Press.

Andrea Cimino and Felice Dell'Orletta. 2016. Building the state-of-the-art in pos tagging of italian tweets. In *CLiC-it/EVALITA*.

Agata Cybulska and Piek Vossen. 2014. Using a sledgehammer to crack a nut? Lexical diversity and event coreference resolution. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC2014)*, Reykjavik, Iceland, May 26-31.

Alexandros Komninos and Suresh Manandhar. 2016. Dependency based embeddings for sentence classification tasks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1490–1500.

LDC. 2005. Ace (automatic content extraction) english annotation guidelines for events ver. 5.4.3 2005.07.01. In *Linguistic Data Consortium*.

Pierre Lison and Jörg Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074. Association for Computational Linguistics.

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.

Anne-Lyse Minard, Manuela Speranza, Eneko Agirre, Itziar Aldabe, Marieke van Erp, Bernardo Magnini, German Rigau, Ruben Urizar, and Fondazione Bruno Kessler. 2015. Semeval-2015 task 4: Timeline: Cross-document event ordering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 778–786.

Paramita Mirza and Anne-Lyse Minard. 2014. Fbk-hlt-time: a complete italian temporal processing system for eventi-evalita 2014. In *Fourth International Workshop EVALITA 2014*, pages 44–49.

Tim O'Gorman, Kristin Wright-Bettner, and Martha Palmer. 2016. Richer event description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of the 2nd Workshop on Computing News Storylines (CNS 2016)*, pages 47–56. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

James Pustejovsky, José M Castano, Robert Ingria, Roser Sauri, Robert J Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R Radev. 2003. Timeml: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34.

James Pustejovsky, José Castao, Robert Ingria, Roser Saurì, Robert Gaizauskas, Andrea Setzer, and Graham Katz. 2003a. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *Fifth International Workshop on Computational Semantics (IWCS-5)*.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348, Copenhagen, Denmark, September. Association for Computational Linguistics.

Sara Tonelli, Alessio Palmero Aprosio, and Marco Mazzon. 2017. The impact of phrases on italian lexical simplification. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017)*, Rome, Italy.

N. UzZaman, H. Llorens, L. Derczynski, J. Allen, M. Verhagen, and J. Pustejovsky. 2013. SemEval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Proceedings of SemEval-2013*, pages 1–9. Association for Computational Linguistics, Atlanta, Georgia, USA.

M. Verhagen, R. Gaizauskas, F. Schilder, M. Hepple, G. Katz, and J. Pustejovsky. 2007. SemEval-2007 Task 15: TempEval Temporal Relation Identification. In *Proceedings of SemEval 2007*, pages 75–80, June.

Marc Verhagen, Roser Sauri, Tommaso Caselli, and James Pustejovsky. 2010. Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 57–62. Association for Computational Linguistics.

Piek Vossen, Tommaso Caselli, and Yiota Kontzopoulou. 2015. Storylines for structuring massive streams of news. In *Proceedings of the First Workshop on Computing News Storylines*, pages 40–49.

Enhancing the Latin Morphological Analyser LEMLAT with a Medieval Latin Glossary

Flavio Cecchini*, Marco Passarotti*, Paolo Ruffolo*, Marinella Testori*,
Lia Draetta°, Martina Fieromonte°, Annarita Liano°, Costanza Marini°, Giovanni Piantanida°
*Università Cattolica del Sacro Cuore - °Università di Pavia
*Largo Gemelli 1, 20123 Milan, Italy - °Corso Strada Nuova 65, 27100 Pavia, Italy
{flavio.cecchini, marco.passarotti}@unicatt.it

Abstract

English. We present the process of expanding the lexical basis of the Latin morphological analyser LEMLAT with the entries from the Medieval Latin glossary Du Cange. This process is performed semi-automatically by exploiting the morphological properties of lemmas, a previously available word list enhanced with inflectional information, and the contents of the lexical entries of Du Cange.

Italiano. *L'articolo descrive il processo di ampliamento della base lessicale dell'analizzatore morfologico per il latino* LEMLAT *con il glossario di latino medievale Du Cange. Il processo è realizzato semiautomaticamente ricorrendo ad alcune proprietà morfologiche dei lemmi, a un lemmario completo d'informazione flessionale e ai contenuti delle entrate lessicali del Du Cange.*

1 Introduction

Latin raises particular challenges for Natural Language Processing (NLP). Given that accuracy rates of stochastic NLP tools heavily depend on the training set on which their models are built, this becomes a particularly problematic issue when Latin is concerned, because Latin texts show an enormous linguistic variety resulting from (a) a wide time span (covering more than two millennia), (b) a large set of genres (ranging from literary to philosophical, historical and documentary texts) and (c) a big diatopic diversity (spread all over Europe and beyond).

Such complexity impacts NLP to the point that building NLP tools claiming to be suitable for all Latin varieties is an unrealistic task. One practical example comes from an experiment described by Ponti and Passarotti (2016), who show that the performance of a dependency parser trained on Medieval Latin data drops dramatically when the same trained model is applied to texts from the Classical era.

This issue affects all layers of linguistic annotation, including fundamental ones, like lemmatisation and morphological analysis. Today, a handful of morphological analysers are available for Latin, chiefly Words,[1] LEMLAT 3.0,[2] Morpheus[3] –reimplemented in 2013 as Parsley[4]–, the PROIEL Latin morphology system[5] and LatMor.[6]

Although LEMLAT, together with LatMor,[7] has proved to be the best performing morphological analyser for Latin and the one boasting the largest lexical basis, its lexical coverage is still limited to Classical and Late Latin only. First released as a morphological lemmatiser at the end of the 1980s at ILC-CNR in Pisa (Bozzi and Cappelli, 1990; Marinone, 1990, v 1.0), where it was enhanced with morphological features between 2002 and 2005 (Passarotti, 2004, v 2.0), LEMLAT relies on a lexical basis resulting from the collation of three Latin dictionaries (Georges and Georges, 1913 1918; Glare, 1982; Gradenwitz, 1904) for a total of 40 014 lexical entries and 43 432 lemmas, as more than one lemma can be included in one lexical entry. This lexical basis was further enlarged in version 3.0 of LEMLAT by semi-automatically adding most of the Onomasticon (26 415 lemmas out of 28 178) provided by the 5th edition of the Forcellini dictionary (Budassi and

[1] http://archives.nd.edu/words.html
[2] www.lemlat3.eu. Binaries and database available at https://github.com/CIRCSE/LEMLAT3.
[3] https://github.com/tmallon/morpheus
[4] https://github.com/goldibex/parsley-core
[5] https://github.com/mlj/proiel-webapp/tree/master/lib/morphology
[6] http://cistern.cis.lmu.de
[7] For an evaluation of morphological analysers for Latin see (Springmann et al., 2016).

Passarotti, 2016).

In order to equip LEMLAT to process Latin texts beyond the Classical period, we recently enhanced its lexical basis with the lexical entries from a large reference glossary for Medieval Latin, namely the *Glossarium Mediae et Infimae Latinitatis* by Du Cange et alii (1883 1887, hereafter DC). This paper details the process performed to include DC in LEMLAT's lexical basis.

2 Word Form Analysis in LEMLAT

LEMLAT is a lemmatiser and morphological analyser of types (i. e. no contextual disambiguation is performed). Given a word form in input (e. g. *coniugae*), LEMLAT's output produces the corresponding lemma(s) (e. g. *coniuga* 'wife') and a number of tags conveying (a) the inflectional paradigm of the lemma(s) (e. g. first declension noun) and (b) the morphological features of the input word form (e. g. feminine singular genitive and dative; feminine plural nominative and vocative).

LEMLAT makes use of a database that includes multiple tables recording the different formative elements (segments) of word forms. The core table is the lexical look-up table, whose basic component is the so-called LES (LExical Segment). The LES is defined as the invariable part of the inflected form (e. g. *coniug* for *coniug-ae*). In other words, the LES is the string (or one of the strings) of characters that remains the same in the inflectional paradigm of a lemma; hence, the LES does not necessarily correspond to either the word stem or the root.

LEMLAT includes a LES archive, in which LES are assigned an ID and a number of inflectional features, among which a tag for the gender of the lemma (for nouns only) and a code (called CODLES) for its inflectional category. According to the CODLES, the LES is compatible with the endings (called SF, "Final Segment") of its inflectional paradigm, which are collected in a separate table in the LEMLAT database. For example, the CODLES for the LES *coniug* is N1 (first declension nouns) and its gender is F (feminine). The word form *coniugae* is thus analysed as belonging to the LES *coniug*, the segment *ae* being recognised as an ending compatible with a LES with CODLES N1.

3 Adding the Du Cange Glossary

Adding DC to LEMLAT is a challenging task mostly because DC is not a dictionary in the modern sense of the word, but a glossary, i. e. a mere collection of words where information about parts of speech (PoS) and inflectional categories is almost absent, and therefore has to be deduced or reconstructed before an entry can be included in LEMLAT.[8] In addition, lemmatisation criteria are often inconsistent, even for words belonging to the same class (e. g. verbs are cited either by their present active infinitive or by their first person singular present indicative).

This is partly due to the fact that five different authors contributed to the glossary over a period of two centuries (Géraud, 1839), not always coherently with respect to their predecessors. Nonetheless, it is possible to distinguish some recurring patterns, which can be exploited to automatically include in LEMLAT as many of the 85 999 lemmas in DC as possible, or at least to expedite the manual recording of lexical entries.

3.1 Suffixes and Bon's Word List

The preliminary step to extend LEMLAT with DC consists in selecting a set of derivational suffixes that are morphologically-unambiguous in terms of PoS and inflectional category, and hence the set of all lemmas displaying these suffixes. These lemmas require no further analysis for entry in LEMLAT. Examples are *-itas* for feminine imparysillabic third declension nouns, or *-icum* for neuter second declension nouns. On the contrary, suffixes like, e. g. *-anus* or *-atus* are considered morphologically-ambiguous, as they can belong to different PoS (adjective or noun) and/or different inflectional categories (first or fourth declension). In these cases the corresponding lemmas require manual annotation (see Section 3.2). Approximately 30 000 DC lemmas are retrieved and added to LEMLAT in this way.

To extend the automatic acquisition of DC's lemmas, we also take advantage of a list of 71 908 Latin lemmas collected by Bruno Bon from various lexicographic sources and corpora.[9] This list supplies information about inflectional morphology.[10] Of these lemmas, 22 628 are found among

[8] For this work, we use the digital version of DC provided by the École nationale des chartes (Paris). Source data are available in XML format at `http://svn.code.sf.net/p/ducange/code/xml/`. The glossary can be accessed online at `http://ducange.enc.sorbonne.fr/`.

[9] Available at `http://glossaria.eu/outils/lemmatisation/` and presented in (Bon, 2011).

[10] Specifically: PoS; genitive endings of nouns; nominative

those in DC that are not analysed in the preliminary step; and out of these, 21 805 showing a one-to-one correspondence with lemmas in Bon's list are added to LEMLAT with no further check.[11]

3.2 Definitions and Quotations

Each lexical entry in DC comprises (a) the name of the lemma, (b) usually, a short **definition** and (c) possibly one or more **quotations** (taken from explicitly-cited textual sources), where most of the times a form of the lexical entry is capitalised. By making use of all these elements, we automatically assign a PoS and an inflectional category (i. e. a CODLES, in LEMLAT's terms) to the lemma.

In particular, to assess the PoS of a lemma we follow a principle of "lexical osmosis", that is, we assume that a lemma's definition core (see below) will most probably use terms belonging to the same PoS of that lemma. By cross-checking this information with the citation form of the lemma and possibly with its inflected forms in a quotation, we are able to assign it also its inflectional category.

With regard to the **definition**, we take into consideration only its initial part, maximally up to the first quotation; what comes after are mostly more in-depth discussions of the term, secondary interpretations or later interpolations. More precisely, we focus on the **definition's core**, i. e. a short capitalised phrase, enclosed in commas and/or ending with a full-stop, providing a short explanation or paraphrase of the lemma immediately after the lemma itself. Its terms are lemmas in typical quotation form, e. g. the nominative case for nouns. Moreover, the definition's core makes use of a standardised and Classical variety of Latin lexicon so as to be as clear as possible to the reader. This means that most of the terms in a definition's core can also be found in the list of lemmas of LEMLAT 3.0. Of the recognised forms, we retain only those that are univocally assigned only one PoS. We ignore a small set of both function and content words often recurring in definitions (e. g. *pro* 'for' and *omnis* 'all, every'), and discard as noise a set of very common lexicographical annotations and abbreviations (e. g. *Italus* or *Ital.*, *f.* = fortasse, *lib.*, *cap.*).

With regard to **quotations**, we only consider the first one as the most significant. Given the lemma's citation form in DC, we exploit the list of all Latin endings and their agreements with inflectional categories available in LEMLAT's database to construct all of its *a priori* possible inflectional paradigms; of these (partly artificial) forms, we retain only those that allow us to unambiguously discriminate a PoS and/or an inflectional category from the others. For example, the entry for *mansaticus* 'mansion, house' illustrates this method:

> MANSATICUS, **Mansio**, domus. Annal. Bertin. ad ann. 874. tom. 7. Collect. Histor. Franc. pag. 118 : *Inde per Attiniacum et consuetos* ***Mansaticos*** *Compendium adiit* [. . .]

Since the definition's core *mansio* can only be a noun for LEMLAT, we can conclude that *mansaticus* is almost surely a noun too, even if the *-icus* ending tends to be associated with denominal adjectives in Latin. The *-us* ending tells us that *mansaticus* can be either a masculine second or fourth declension noun;[12] a first class adjective might theoretically be possible, but is ruled out by the definition's core *mansio*. The second declension is confirmed by the ending *-os* found in the quotation, thus excluding the fourth declension (which should yield *-us*).

Thanks to this process, more than 10 000 additional lemmas are automatically included in LEMLAT. This process is applied very carefully, covering only decidedly unambiguous cases, i. e. when content words in the definition's core are found to belong to only one PoS or to a phrase of a fixed type (e. g. a phrase ending with an infinitive assigns PoS verb to the lemma) and when the inflectional category of the word form possibly found in the quotation can be univocally discriminated. This leads to high precision (1.0), but affects recall (0.18). For the remaining cases we have to resort to manual annotation; this happens most frequently when we correctly identify the PoS and the inflectional category of a lemma, but cannot infer its gender *a priori*. For instance, approxi-

endings of adjectives; infinitive endings of regular verbs and full paradigms of irregular verbs.

[11] The remaining lemmas are manually-checked because they correspond to multiple entries in one and/or the other source. For example, the lemma *fedus* appears once in DC (as a masculine second declension noun, 'fief') but three times in Bon's list: as a masculine second declension noun (but with the different meaning 'goat'), as a neuter third declension noun (with the genitive *federis*, 'alliance') and as a first class adjective ('hideous').

[12] Feminines are so rare in these declensions that we exclude them from the automated analysis.

mately 10% of first declension nouns are found to be masculine, and not feminine as expected.

4 Discussion

Not all of the 85 999 lemmas of DC are included in LEMLAT. We exclude the entries of some 3 000 fixed or idiomatic multi-word expressions and of around 300 adverbs derived either from an adjective (e. g. *affectuose* 'tenderly' from *affectuosus* 'tender') or from a verb (e. g. *attendenter* 'watchfully' from *attendere* 'to keep, to watch') in the lexical basis of the DC-enhanced LEMLAT. This is because LEMLAT considers derived adverbs as part of the inflectional paradigm of the source adjective or verb.

At the end of the process, 82 556 DC lemmas are added to LEMLAT. Since DC shows a tendency to treat different nuances of the same lemma as distinct entries, the total number of DC distinct lemmas inserted in LEMLAT is 73 131. The lemmas with the highest number of separate entries are *forma* 'form' (17), *scala* 'stairs, staircase, ladder' (15) and *status* 'mode, state, position, size' (15). These are all already attested in Classical Latin, but are also recorded in DC because of their semantic change over time.[13] This happens frequently; there are, in fact, 10 168 shared lemmas (corresponding to 14 469 entries in DC) in LEMLAT 3.0 and DC, with respect to the name of the lemma, its PoS and inflectional category (and gender, when applicable). Additionally, 1 820 lemmas share the same quotation form in both sources (often incidentally), despite being morphologically different. An example is *amo*: in DC, it is the third declension noun *amo*, *amonis*, a variant of *ammo*, *ammonis* (a unit of measure for wine), while in LEMLAT it is the verb *amare* 'to love'.

The remaining 66 267 lemmas are to be considered lexical innovations of "*media et infima Latinitas*". Looking at these Medieval lemmas, we notice some tendencies in the distribution of PoS and inflectional categories. Whereas nouns are the prevalent PoS both in LEMLAT and DC (albeit at very different rates, respectively 52% and 75%), in the former the most attested declension is the third (37% of nouns), while in the latter it is the first and second declensions that dominate (34% and 39% of nouns, against 20% of the third declension), showing a trend towards more transparent lexical items. While similar figures can be observed for verbs, in DC we notice a reduced presence of adjectives (12% against LEMLAT's 25%), revealing that they represent a less diachronically-productive PoS than nouns and verbs.

5 Evaluation

As conducted for the previous major update of LEMLAT (Passarotti et al., 2017), we evaluate LEMLAT's coverage of the Latin lexicon against the *Thesaurus formarum totius latinitatis* (TFTL) by Tombeur (1998), in order to assess the impact of LEMLAT's acquisition of DC. A primary reference for the study of the Latin lexicon, TFTL is a comprehensive diachronic collection of all Latin word forms as they occur in texts from the archaic period up to the Second Vatican Council (20th century), listing their respective frequencies in the sources from different eras.[14]

Passarotti et alii (2017) report a coverage of 72.254% of TFTL's forms, corresponding to 98.345% of the 62 922 781 total occurrences in the source texts.[15] This is partly explained by the fact that many forms in TFTL are either extremely rare, include punctuation in their spelling, or are merely sequences of numbers, letters and punctuation marks. When we add DC to LEMLAT, our coverage of TFTL raises by 3.264% to 75.518%, corresponding to 17 224 newly-recognised forms, whereas the covered occurrences increase to 98.665%.

We also perform a coverage evaluation over three Medieval Latin texts of comparable size, available from ALIM, the Archive of Italian Medieval Latinity (Ferrarini, 2017).[16] The texts belong to three different periods and genres; these are: the *Codex diplomaticus Cavensis* I (documents 33-210), a collection of documentary sources from Southern Italy dating to the 9th century; the *Historia Mongalorum*, a 13th century report of a journey and diplomatic mission; and the *De falso credita et ementita Constantini donatione*, a philological treatise dating back to the end of the 15th century.

[13] Indeed, DC does not at all record lemmas already available in Classical Latin, unless they show a different meaning and/or morphology.

[14] Archaic Latin (up to IInd c. AD), Patristic Latin (IInd c. AD – AD 735), Medieval Latin (AD 736 – AD 1499) and Modern Latin (AD 1500 – AD 1965), respectively.

[15] The statistics in this paper are based on updated, marginally corrected statistics with respect to those presented in Passarotti et alii (2017).

[16] http://it.alim.unisi.it/

Work (century)	Tokens	Types	LEMLAT	LEMLAT + DC	Only DC
Codex dipl. Cavensis (IX)	19428	3262	54.1%	59.2%	166 (5.1%)
Historia Mongalorum (XIII)	20360	4649	90.3%	92.2%	87 (1.9%)
De Constantini donatione (XV)	19805	6514	93.9%	94.8%	56 (0.9%)

Table 1: Comparison of the lexical coverage of DC-enhanced LEMLAT of three Medieval texts. The "Only DC" column lists the number of terms to be found exclusively in the added DC vocabulary.

Table 1 shows the improvements in lexical coverage obtained thanks to the enhancement of LEMLAT through DC. The results are in line with those for TFTL. Remarkably, the highest increase in performance is recorded for the least-standardised of the three texts, the *Codex diplomaticus*, which remains the most demanding for LEMLAT to analyse. This can be explained by the large presence of local names of people and places (e. g. *Sichelpertus*, *Eboli*), and especially by the very frequent deviations from the orthographic standard (e. g. *abentes* for *habentes* 'having (pl.)', *ecclesie* for *ecclesiae* 'of/to the church; churches'); the latter are also the source of many false positives, which LEMLAT does not discriminate from true positives. Names are challenging, too, as can be observed, for example, from the fact that among the 363 unrecognised forms in the *Historia Mongalorum*, the majority are ethnonyms, toponyms and anthroponyms (e. g. *Caracoron* 'Karakorum', *circassos* 'Circassians', *Mengu* 'Möngkh').

At the same time, LEMLAT is now able to analyse words which, while absent from the vocabulary of Classical Latin, are tied to key, widespread concepts in the Middle Ages. For example, in the *Historia Mongalorum* the enhanced LEMLAT can now detect terms like *orda* 'horde' (11 occurrences) or *protonotarius* 'prothonotary' (4 occurrences), both important in the 13th century onward in the context of conflicts and diplomatic missions between Western Europe and the Mongol Empire. Interestingly, the source for these lemmas in DC is not the *Historia Mongalorum* itself, which is an indication of the effective circulation of such words.

6 Conclusion

In this paper we present the rule-based process performed to semi-automatically enhance the Latin morphological analyser LEMLAT with the Du Cange glossary. While dated, such an approach is still necessary if the intent is to minimise the error rate resulting from the automatic PoS-tagging of the glossary's definitions and quotations. Indeed, unless tuned on an in-domain training set, existing stochastic PoS-taggers for Latin are not yet reliable enough when it comes to processing the complex, raw and "freestyle" definitions of DC.

The ever-growing availability of digitised Latin texts from various eras urges us to build NLP tools capable of automatically analysing such varied sets of linguistic data. In this respect, enhancing the lexical basis of LEMLAT with a Medieval Latin dictionary is a first step towards the development of well-performing tools on diachronic data. Conversely, even if building a tool suitable for different diachronic varieties of Latin were feasible for low-level annotation tasks (like e. g. lemmatisation and morphological analysis), this does not seem to be the case for tasks such as syntactic parsing or word sense disambiguation, for which either highly flexible or highly specialised tools will be needed.

This is an open issue not only for Latin. Indeed, the portability of NLP tools across domains and genres is currently one of the main challenges in NLP. Thanks to its highly diverse corpus, Latin is a perfect case-study language to tackle these problems.

For the future, we plan to expand LEMLAT's lexical database with all of the graphical variants reported in DC and possibly also with other Medieval Latin thesauri, such as the *Dictionary of Medieval Latin from British Sources* (Ashdown et al., 2018), so as to improve both its diatopic and diachronic coverage. In general, we aspire to make LEMLAT's algorithm better able to cope with the most widespread and predictable orthographic variations recorded in Medieval manuscripts and texts.[17]

[17] An introduction and an approach to this issue can be found in Kestemont and De Gussem (2017).

References

Richard K Ashdown, David R Howlett, and Ronald E Latham, editors. 2018. *Dictionary of Medieval Latin from British Sources*. Oxford University Press for the British Academy, Oxford, UK.

Bruno Bon. 2011. OMNIA : outils et méthodes numériques pour l'interrogation et l'analyse des textes médiolatins (3). *Bulletin du centre d'études médiévales d'Auxerre* BUCEMA, (15). Online at `http://journals.openedition.org/cem/12015`.

Andrea Bozzi and Giuseppe Cappelli. 1990. A project for Latin lexicography: 2. A Latin morphological analyzer. *Computers and the Humanities*, 24(5-6):421–426.

Marco Budassi and Marco Passarotti. 2016. Nomen omen. Enhancing the Latin morphological analyser Lemlat with an onomasticon. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH)*, pages 90–94, Berlin, Germany. Association for Computational Linguistics.

Charles du Fresne du Cange, Bénédictins de Saint-Maur, Pierre Carpentier, Louis Henschel, and Léopold Favre. 1883–1887. *Glossarium mediae et infimae latinitatis*. Niortm France.

Edoardo Ferrarini. 2017. ALIM ieri e oggi. *Umanistica Digitale*, 1(1). Online at `https://umanisticadigitale.unibo.it/article/view/7193`.

Karl Ernst Georges and Heinrich Georges. 1913–1918. *Ausführliches lateinisch-deutsches Handwörterbuch*. Hahn, Hannover, Germany.

Hercule Géraud. 1839. Historique du glossaire de la basse latinité de Du Cange. *Bibliothèque de l'École Nationale des Chartes*, 1:498–510.

Peter GW Glare. 1982. *Oxford Latin dictionary*. Clarendon Press. Oxford University Press, Oxford, UK.

Otto Gradenwitz. 1904. *Laterculi Vocum Latinarum: voces Latinas et a fronte et a tergo ordinandas*. Hirzel, Leipzig, Germany.

Mike Kestemont and Jeroen De Gussem. 2017. Integrated Sequence Tagging for Medieval Latin Using Deep Representation Learning. *Journal of Data Mining & Digital Humanities*, Special Issue on Computer-Aided Processing of Intertextuality in Ancient Languages, August. Online at `https://jdmdh.episciences.org/3835`.

Nino Marinone. 1990. A project for Latin lexicography: 1. Automatic lemmatization and word-list. *Computers and the Humanities*, 24(5-6):417–420.

Marco Passarotti, Marco Budassi, Eleonora Litta, and Paolo Ruffolo. 2017. The 3.0 Package for Morphological Analysis of Latin. In *Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language*, pages 24–31, Gothenburg, Sweden. Northern European association for language technology (NEALT), Linköping University Electronic Press.

Marco Passarotti. 2004. Development and perspectives of the Latin morphological analyser LEMLAT. *Linguistica computazionale*, XX-XXI:397–414.

Edoardo Maria Ponti and Marco Passarotti. 2016. Differentia compositionem facit. a slower-paced and reliable parser for Latin. In *Proceedings of the tenth international Conference on Language Resources and Evaluation (*LREC '16*)*, pages 683–688, Portorož, Slovenia. European Language Resources Association (ELRA).

Uwe Springmann, Helmut Schmid, and Dietmar Najock. 2016. LatMor: A Latin finite-state morphology encoding vowel quantity. *Open Linguistics - Topical Issue on Treebanking and Ancient Languages: Current and Prospective Research*, 2(1):386–392.

Paul Tombeur. 1998. *Thesaurus formarum totius Latinitatis: a Plauto usque ad saeculum XXum*. Brepols, Turnhout, Belgium.

Is Big Five better than MBTI?
A personality computing challenge using Twitter data

Fabio Celli
FBK - MobS and Profilio Company
Trento, Italy
celli@fbk.eu

Bruno Lepri
FBK - MobS
Trento, Italy
lepri@fbk.eu

Abstract

English. Personality Computing from text has become popular in Natural Language Processing (NLP). For assessing gold-standard personality types, Big5 and MBTI are two popular models but still there is no comparison of the two in personality computing. With this paper, we provide for the first time a comparison of the two models from a computational perspective. To do that we exploit two multilingual datasets collected from Twitter in English, Italian, Spanish and Dutch.

Italiano. *Il riconoscimento automatico di personalità è diventato popolare nelle comunità di linguistica computazionale. I test Big Five e MBTI sono due modelli differenti per valutare la personalità, ma ancora non c'è un vero confronto dei due in ambito di riconoscimento automatico di personalità. In questo articolo per la prima volta forniamo una comparazione dei due modelli dal punto di vista computazionale. Per fare questo abbiamo raccolto dati Twitter in Inglese, Italiano, Spagnolo e Olandese in due corpora paralleli annotati con i due test.*

1 Introduction

The last decade has been characterized by the rise of personality computing in Natural Language Processing (NLP) (Vinciarelli and Mohammadi, 2014): for example, several works have dealt with the automatic prediction of personality traits of authors from different pieces of text they wrote in emails, blogs or social media (Mairesse et al., 2007; Iacobelli et al., 2011; Schwartz et al., 2013) (Rangel Pardo et al., 2015). Personality computing is also broadening its application to many fields in academia as well as in industry, including security (Golbeck et al., 2011), human resources (Turban et al., 2017), advertising (Celli et al., 2017) and deception detection (Fornaciari et al., 2013). Historically, there are two popular but very different psychological tests to asses personality: (i) the Big Five (Costa and McCrae, 1985; Costa and McCrae, 2008), which is widely accepted in academia, and (ii) the Myers Briggs Type Indicator (MBTI) (Myers and Myers, 2010), which is very popular and widely used in industry. The Big Five model defines personality along 5 bipolar scales: Extraversion (sociable vs. shy); Emotional Stability (secure vs. neurotic); Agreeableness (friendly vs. ugly); Conscientiousness (organized vs. careless); Openness to Experience (insightful vs. unimaginative). In contrast, the MBTI defines 4 binary classes that combines into 16 personality types: Extraversion/Introversion, Sensing/Intuition, Perception/Judging, Feeling/Thinking. Correlation analyses of the personality measures showed that Big Five Extraversion was correlated with MBTI Extraversion-Introversion, Openness to Experience was correlated with Sensing-Intuition, Agreeableness with Thinking-Feeling and Conscientiousness with Judging-Perceiving (Furnham et al., 2003). A reason for the recently gained popularity of MBTI is the fact that it is easier to collect gold-standard labelled data about MBTI than about Big Five, as an MBTI type is a 4-letter coding (e.g., INTJ) that could be retrieved with simple queries. In a field like personality computing, where data is costly and difficult to collect, this is an enormous advantage.

In this paper we address the question whether it is easier to predict Big Five or MBTI classes with a machine learning approach. To do so, we collect two Twitter datasets in English, Italian, Dutch and Spanish, one annotated with the Big Five personality types and one with MBTI. We believe that this

work will be useful for the scientific community of personality computing to better understand the heuristic power of the two models when applied to machine learning tasks.
The paper is structured as follows: in the next section we provide an overview of related works in the field of personality computing in NLP, in Section 3 we describe the datasets we used, in Section 4 we report the results of our experiments and in Section 5 we draw some conclusions.

2 Related Work

Brief overview of personality computing The research in personality computing from text begun more than a decade ago with few pioneering works recognizing personality traits (Big Five traits) from blogs (Oberlander and Nowson, 2006) and self presentations (Mairesse et al., 2007). Other related fields have developed in the same years, like personality computing from multimodal and social signals, such as recorded meetings (Pianesi et al., 2008). In that period the research on MBTI was limited to find correlates between personality types and behavioral expectations, such as job preference (Cohen et al., 2013). Thus, MBTI was marginally used for personality computing until 2015 (Luyckx and Daelemans, 2008); while many works demonstrated the validity of Big Five for the automatic prediction of personality from different sources, including Twitter (Quercia et al., 2011) (Pratama and Sarno, 2015) (Qiu et al., 2012). The most common features used by researchers to perform such tasks were extracted from text, such as sentiment (Basile and Nissim, 2013), Part of Speech (PoS) tags, psycholinguistic tags (LIWC) (Tausczik and Pennebaker, 2010) and from metadata, such as number of followers, density of subject's network, hashtags, Likes and profile pictures. The rise of personality computing by means of the Big Five model brought fruitful collaborations between the communities of computer science and personality psychology (Back et al., 2010), and very interesting findings came out: for example that several personal characteristics extracted from social media profiles such as education, religion, marital status and the number of political preferences have really high correlations with personality types (Kosinski et al., 2013), or that popular users in social media are both extroverts and emotionally stable as well as high in Openness, while influential ones tend to be high in Conscientiousness (Quercia et al., 2012).

Overview of datasets The scarcity of data annotated with gold standard personality labels, difficult and costly to collect, was a major problem and the few large datasets available (MyPersonality, about 75K users, and Essays, about 2K users) soon became standard benchmarks (Celli et al., 2013). These available datasets covered mainly English language, while all the other datasets were much smaller, around 200 or 300 instances. In this scenario a dataset of 1500 instances collected by means of a simple Twitter search came out, and it was in English and annotated with MBTI labels (Plank and Hovy, 2015). This demonstrated that MBTI labels are very common and easy to retrieve from Twitter, unlike Big Five labels. Soon thereafter, TwiSty came out (Verhoeven et al., 2016), a multilanguage dataset of 17K instances annotated with MBTI and including Italian, Dutch, Portuguese, French and Spanish.

State of the art The MBTI model formalizes personality types as classes, while Big Five as scores. Despite this, works in computer science and computational linguistics split between those who use scores (Golbeck et al., 2011) and those who turn Big Five scores into binary classes in order to have a better control on class distribution and easier-to-interpret prediction tasks (Mairesse et al., 2007) (Segalin et al., 2017). In particular, Mairesse et al. obtained an average of 57% accuracy in the prediction of Big Five classes using the LIWC psycholinguistic features, also reporting that Openness to Experience was the easiest trait to model. Verhoeven et al. (Verhoeven et al., 2013) obtained a 72% of F-measure in the prediction of Big Five using trigrams and ensemble methods in a small Facebook dataset trained on a larger essays dataset. In a following study, Verhoeven et al. (Verhoeven et al., 2016) obtained an average of 63.8% of F-measure in the prediction of MBTI on Twitter in multiple languages using word and characters n-grams. Again, Farnadi et al. (Farnadi et al., 2013) obtained an average accuracy of 58.6% to predict Big Five classes on the same dataset using mostly metadata. Finally, Plank and Hovy (Plank and Hovy, 2015) used words and Twitter metadata to predict Extraversion/Introversion and Feeling/Thinking with 72% and 61% of accuracy, respectively. They reported that the best performing features are the linguistic ones.

The different settings and datasets used by previous works in the field makes it impossible to compare the results. Here, we aim to fill this gap.

3 Datasets

We collected from Twitter two multilingual datasets, of 900 users each, one annotated with MBTI and one with Big Five. First we collected the Big Five set by means of queries with Twitter advanced search[1], retrieving the results of different Big Five tests, ranging from the short 10-items test to the 44-items test. The language of the tweets were English, Italian, Spanish and Dutch, so we replicated the language distribution in the MBTI set using a portion of TwiSty (Verhoeven et al., 2016) and Plank's corpus (Plank and Hovy, 2015). The details about language distributions are reported in Figure 1.

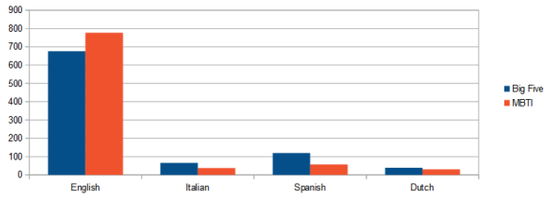


Figure 1: Distribution of the languages in the two datasets. The x-axis represents the number of users.

As expected there are many more tweets containing the results of the MBTI with respect to the Big Five. We use a concatenation of all tweets of a user, and a limit to 40 tweets per user in order to balance those who have too many tweets those that have few. In the end we used two comparable datasets with 900 users each, 265K words in the Big Five one and 290K words in the MBTI one. The classes are balanced in the Big Five set, as we obtained them with a median split from the original scores, on the contrary in the MBTI set there is a strong imbalance in the distribution of Sensing/Intuition and Feeling/Thinking, reported also in Plank's corpus. In the experiments, described in the next section, we balance the classes of both datasets and test different combinations of the features to evaluate the performance of machine leaning algorithms in the prediction of classes derived from the two different personality models.

[1] https://twitter.com/search-advanced

4 Experiments, Results, Discussion and Limitations

Experimental settings We compared the performance of algorithms for the prediction of Big Five and MBTI classes in 9 binary classification tasks. To do so, we used the following features:

- Character n-grams (1000 features): we extracted from tweets 1000 characters bi-grams and tri-grams with a minimum frequency of 3. We did not remove stopwords and punctuation;

- LIWC match ratio (68 features): we computed the ratio of matches of the words in the LIWC dictionaries in all the four languages. LIWC provides mapping from words to 68 psycholinguistic categories, including words about others, self, space, time, society, family, friendship, sex, and functional words, among others;

- Metadata (10 features): this feature set includes the followers/following ratio, favorite/tweets ratio, listed/tweets ratio, link color, text color, border color, background color, hashtag/words ratio, retweet ratio, whether the profile picture is the default one or not. As feature selection procedure we used a subset selection algorithm (Hall and Smith, 1998) that reduces the degree of redundancy. We balanced the classes assigning weights to the instances in the data so that each class has the same total weight. For the classification we compared SVMs and a meta-classifier that automatically finds the best performing algorithm for the task (Thornton et al., 2013). As evaluation setting we used a 10-fold cross validation, as metric we reported accuracy and averages. For the maximum comparability we also reported the average on the Big Five four traits correlated with MBTI (avg4): extraversion, openness, agreableness and conscientiousness.

Results and discussion Results reported in Table 1 show that, on average, SVMs have higher performance in the prediction of MBTI classes with respect to Big Five, but there is much variability in the prediction of Big Five traits. In particular, we obtained very good performances for Emotional Stability and Agreeableness using a SVMs with polynomial kernel and Random Sub Spaces respectively, but poor with simple SVMs, indicating that the space is not linearly separable. On the contrary, the predictions of the MBTI seems to be more stable, in contrast to the results of Plank and Hovy. We suggest that this different

trait	baseline	svm	auto	best feature
extr.	49.6	61.8	66.4 lr	others
stab.	49.8	59.6	74.8 svmk	I
agree.	49.6	61.1	73.3 rss	death
consc.	49.8	60.3	61.6 sdg	death
open.	49.6	53.1	59.4 nb	ngrams
avg4	49.7	**59.0**	**65.1**	-
avg	49.7	59.1	67.0	-
E-I	49.5	63.9	64.7 sdg	hashratio
S-N	49.2	66.3	68.6 bag	negate
F-T	49.8	63.0	63.0 svm	self
P-J	49.5	61.7	63.5 nb	self
avg	49.5	**63.7**	**64.9**	-

Table 1: Results of the experiments with all the languages and 900 instances per each set. Big Five is in the upper part of the Table and MBTI is below. We report accuracies for Support Vector Machines (svm) and AutoWeka (auto), a meta-classifier that automatically finds the best algorithm and settings for the task. The auto meta-classifier used Logistic Regression (lr), Support Vector Machines with polynomial kernel (svmk), Random Sub Spaces (rss), Stochastic Gradient Descent Regression (sdg), Naive Bayes (nb) and Bagging (bag). We also report average accuracy of Big Five traits correlated to MBTI (avg4): Extraversion, Openness to Experience, Agreeableness and Conscientiousness. The best features for the predictions are: words about others (others), first person singular pronoun (I), words about death (death), ngrams (ngrams), words about self (self), negation words (negate), hashtag ratio (hashratio).

result is due to three factors: class balancing, the use of LIWC and the subset feature selection. It is interesting to note that the *reference to others* is the best feature for the prediction of Big Five Extraversion and *first person pronouns* for the prediction of Emotional Stability/Neuroticism. We explain the predictive power of words about death for Agreeableness and Conscientiousness with the fact that this feature is correlated to the negative poles of these traits. The presence of different languages might affect negatively the performance so we ran an experiment using only English (650 users for each set).

Results, reported in Table 2, show that the effect of language variety is minimum, given that English is the most represented language in the datasets. It is interesting to note the changes in the best features: *hashtag ratio* is in English the best feature for Extraversion Big Five, while in the previous experiment it was the best feature for Extraversion MBTI. Here the best feature for Extraversion MBTI is *anger*, that is a clue for the negative class of this trait: Introversion. It is also interesting to note that words about feelings become in English the best feature for Agreeableness, although the performance decreases a little bit with respect to the experiment with all languages.

trait	baseline	svm	best feature
extr.	49.6	66.1	hashratio
stab.	49.6	62.9	I
agree.	49.6	59.7	feel
consc.	49.4	60.2	ngrams
open.	49.5	60.3	ngrams
avg4	49.6	**61.5**	-
avg	49.6	61.8	-
E-I	49.7	61.3	anger
S-N	48.4	68.5	we
F-T	49.3	68.6	self
P-J	49.6	60.2	I
avg	49.5	**64.6**	-

Table 2: Results of the experiments with English only and 650 instances per each set. Big Five is in the upper part of the Table and MBTI is below. We report accuracy for the majority baseline and Support Vector Machines (svm). The best features for the predictions are: hashtag ratio (hashratio), first person singular pronoun (I), words about feelings (feel), ngrams (ngrams), words about self (self), negation words (negate), words about anger (anger), first person plural pronoun (we), words about self (self).

Limitations In order to compare the two personality models, we forced the Big Five outcome, originally scores, into classes. This is one of the reasons why it is more difficult to predict Big Five classes than MBTI, but it is interesting to note that the performance of some Big Five traits can be boosted using non-linear models. Another limitation is related to the fact that we collected different users in the two datasets, with the risk to have some individuals in one dataset or the other that are easier to classify. In any case, it is impossible to collect data of the same users annotated with both MBTI and Big Five with Twitter queries, this is something that could be done only with a costly data collection effort, that we hope future work will do.

5 Conclusion

In this paper we provide for the first time a comparison of Big Five and MBTI from a personality computing perspective. To do so we use two multilingual Twitter datasets, one annotated with Big Five classes and one with MBTI classes. For the first time, we provide an evidence that algorithms trained on MBTI could have better performances than trained on the Big Five, although the Big Five is much more informative and has great variability in performance depending also on the algorithm used for the prediction. We let available the files used for the experiments[2], in order to grant the replicability or improvement of the results.

[2]http://personality.altervista.org/fabio.htm

Acknowledgments

The work of Fabio Celli and Bruno Lepri was partly funded by EIT Digital by City Enabler for Digital Urban Services (CEDUS) and by EIT Distributed Ledger Invoice (DLI).

References

Mitja D Back, Juliane M Stopfer, Simine Vazire, Sam Gaddis, Stefan C Schmukle, Boris Egloff, and Samuel D Gosling. 2010. Facebook profiles reflect actual personality, not self-idealization. *Psychological science*.

Valerio Basile and Malvina Nissim. 2013. Sentiment analysis on italian tweets. *WASSA 2013*, page 100.

Fabio Celli, Fabio Pianesi, David Stillwell, and Michal Kosinski. 2013. Workshop on computational personality recognition: Shared task. In *WCPR in conjuction to ICWSM 2013*.

Fabio Celli, Pietro Zani Massani, and Bruno Lepri. 2017. Profilio: Psychometric profiling to boost social media advertising. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 546–550. ACM.

Yuval Cohen, Hana Ornoy, and Baruch Keren. 2013. Mbti personality types of project managers and their success: A field survey. *Project Management Journal*, 44(3):78–87.

Paul T Costa and Robert R McCrae. 1985. *The NEO personality inventory: Manual, form S and form R*. Psychological Assessment Resources.

Paul T Costa and Robert R McCrae. 2008. The revised neo personality inventory (neo-pi-r). *In G.J. Boyle, G Matthews and D. Saklofske (Eds.). The SAGE handbook of personality theory and assessment*, 2:179–198.

Golnoosh Farnadi, Susana Zoghbi, Marie-Francine Moens, and Martine De Cock. 2013. Recognising personality traits using facebook status updates. In *Proceedings of the workshop on computational personality recognition (WCPR13) at the 7th international AAAI conference on weblogs and social media (ICWSM13)*. AAAI.

Tommaso Fornaciari, Fabio Celli, and Massimo Poesio. 2013. The effect of personality type on deceptive communication style. In *Intelligence and Security Informatics Conference (EISIC), 2013 European*, pages 1–6. IEEE.

Adrian Furnham, Joanna Moutafi, and John Crump. 2003. The relationship between the revised neo-personality inventory and the myers-briggs type indicator. *Social Behavior and Personality: an international journal*, 31(6):577–584.

Jennifer Golbeck, Cristina Robles, Michon Edmondson, and Karen Turner. 2011. Predicting personality from twitter. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pages 149–156. IEEE.

Mark A Hall and Lloyd A Smith. 1998. *Practical feature subset selection for machine learning*. Springer.

Francisco Iacobelli, Alastair J Gill, Scott Nowson, and Jon Oberlander. 2011. Large scale personality classification of bloggers. In *Affective Computing and Intelligent Interaction*, pages 568–577. Springer.

Michal Kosinski, David Stillwell, and Thore Graepel. 2013. Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the National Academy of Sciences*, 110(15):5802–5805.

Kim Luyckx and Walter Daelemans. 2008. Personae: a corpus for author and personality prediction from text. In *Proceedings of the 6th International Conference on Language Resources and Evaluation. Marrakech, Morocco: European Language resources Association*.

François Mairesse, Marilyn A Walker, Matthias R Mehl, and Roger K Moore. 2007. Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research*, 30(1):457–500.

Isabel Briggs Myers and Peter B Myers. 2010. *Gifts differing: Understanding personality type*. Davies-Black Publishing.

Jon Oberlander and Scott Nowson. 2006. Whose thumb is it anyway?: classifying author personality from weblog text. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 627–634. Association for Computational Linguistics.

Fabio Pianesi, Nadia Mana, Alessandro Cappelletti, Bruno Lepri, and Massimo Zancanaro. 2008. Multimodal recognition of personality traits in social interactions. In *Proceedings of the 10th international conference on Multimodal interfaces*, pages 53–60. ACM.

Barbara Plank and Dirk Hovy. 2015. Personality traits on twitter - or - how to get 1,500 personality tests in a week. *6TH Workshop on computational approaches to subjectivity, sentiment and social media analysis WASSA 2015*, page 92.

Bayu Yudha Pratama and Riyanarto Sarno. 2015. Personality classification based on twitter text using naive bayes, knn and svm. In *Data and Software Engineering (ICoDSE), 2015 International Conference on*, pages 170–174. IEEE.

Lin Qiu, Han Lin, Jonathan Ramsay, and Fang Yang. 2012. You are what you tweet: Personality expression and perception on twitter. *Journal of Research in Personality*, 46(6):710–718.

Daniele Quercia, Michal Kosinski, David Stillwell, and Jon Crowcroft. 2011. Our twitter profiles, our selves: Predicting personality with twitter. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pages 180–185. IEEE.

Daniele Quercia, Renaud Lambiotte, David Stillwell, Michal Kosinski, and Jon Crowcroft. 2012. The personality of popular facebook users. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pages 955–964. ACM.

Francisco Manuel Rangel Pardo, Fabio Celli, Paolo Rosso, Martin Potthast, Benno Stein, and Walter Daelemans. 2015. Overview of the 3rd author profiling task at pan 2015. In *Cappellato L., Ferro N., Jones G., San Juan E. (Eds.) CLEF 2015 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings. CEUR-WS.org, vol. 1391*, pages 1–8.

Andrew H Schwartz, Johannes C Eichstaedt, Margaret L Kern, Lukasz Dziurzynski, Stephanie M Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin EP Seligman, et al. 2013. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one*, 8(9):773–791.

Cristina Segalin, Fabio Celli, Luca Polonio, Michal Kosinski, David Stillwell, Nicu Sebe, Marco Cristani, and Bruno Lepri. 2017. What your facebook profile picture reveals about your personality. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 460–468. ACM.

Yla R Tausczik and James W Pennebaker. 2010. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54.

Chris Thornton, Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. 2013. Auto-weka: Combined selection and hyperparameter optimization of classification algorithms. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 847–855. ACM.

Daniel B Turban, Timothy R Moake, Sharon Yu-Hsien Wu, and Yu Ha Cheung. 2017. Linking extroversion and proactive personality to career success: The role of mentoring received and knowledge. *Journal of Career Development*, 44(1):20–33.

Ben Verhoeven, Walter Daelemans, and Tom De Smedt. 2013. Ensemble methods for personality recognition. In *Proc of Workshop on Computational Personality Recognition, AAAI Press, Melon Park, CA*, pages 35–38.

Ben Verhoeven, Walter Daelemans, and Barbara Plank. 2016. Twisty: a multilingual twitter stylometry corpus for gender and personality profiling. In *Proceedings of the 10th Annual Conference on Language Resources and Evaluation (LREC 2016)/Calzolari, Nicoletta [edit.]; et al.*, pages 1–6.

Alessandro Vinciarelli and Gelareh Mohammadi. 2014. A survey of personality computing. *IEEE Transactions on Affective Computing*, 5(3):1–1.

Automatically Predicting User Ratings for Conversational Systems

A. Cervone[1], **E. Gambi**[1], **G. Tortoreto**[2], **E.A. Stepanov**[2], **G. Riccardi**[1]
[1]Signals and Interactive Systems Lab, University of Trento, Trento, Italy
[2] VUI, Inc., Trento, Italy
{alessandra.cervone, enrico.gambi, giuseppe.riccardi}@unitn.it,
{eas,gtr}@vui.com

Abstract

English. Automatic evaluation models for open-domain conversational agents either correlate poorly with human judgment or require expensive annotations on top of conversation scores. In this work we investigate the feasibility of learning evaluation models without relying on any further annotations besides conversation-level human ratings. We use a dataset of rated (1-5) open domain spoken conversations between the conversational agent Roving Mind (competing in the Amazon Alexa Prize Challenge 2017) and Amazon Alexa users. First, we assess the complexity of the task by asking two experts to re-annotate a sample of the dataset and observe that the subjectivity of user ratings yields a low upper-bound. Second, through an analysis of the entire dataset we show that automatically extracted features such as user sentiment, Dialogue Acts and conversation length have significant, but low correlation with user ratings. Finally, we report the results of our experiments exploring different combinations of these features to train automatic dialogue evaluation models. Our work suggests that predicting subjective user ratings in open domain conversations is a challenging task.

Italiano. *I modelli stato dell'arte per la valutazione automatica di agenti conversazionali open-domain hanno una scarsa correlazione con il giudizio umano oppure richiedono costose annotazioni oltre al punteggio dato alla conversazione. In questo lavoro investighiamo la possibilità di apprendere modelli di valutazione attraverso il solo utilizzo di punteggi umani dati all'intera conversazione. Il corpus utilizzato è composto da conversazioni parlate open-domain tra l'agente conversazionale Roving Mind (parte della competizione Amazon Alexa Prize 2017) e utenti di Amazon Alexa valutate con punteggi da 1 a 5. In primo luogo, valutiamo la complessità del task assegnando a due esperti il compito di riannotare una parte del corpus e osserviamo come esso risulti complesso perfino per annotatori umani data la sua soggettività. In secondo luogo, tramite un'analisi condotta sull'intero corpus mostriamo come features estratte automaticamente (sentimento dell'utente, Dialogue Acts e lunghezza della conversazione) hanno bassa, ma significativa correlazione con il giudizio degli utenti. Infine, riportiamo i risultati di esperimenti volti a esplorare diverse combinazioni di queste features per addestrare modelli di valutazione automatica del dialogo. Questo lavoro mostra la difficoltà del predire i giudizi soggettivi degli utenti in conversazioni senza un task specifico.*

1 Introduction

We are currently witnessing a proliferation of conversational agents in both industry and academia. Nevertheless, core questions regarding this technology remain to be addressed or analysed in greater depth. This work focuses on one such question: *can we automatically predict user ratings of a dialogue with a conversational agent?*

Metrics for task-based systems are generally related to the successful completion of the task. Among these, contextual appropriateness (Danieli and Gerbino, 1995) evaluates, for example, the degree of contextual coherence of machine turns with respect to user queries which are classified with ternary values for slots (appropriate, inappro-

priate, and ambiguous). The approach is somewhat similar to the attribute-value matrix of the popular PARADISE dialog evaluation framework (Walker et al., 1997), where there are matrices representing the information exchange requirements between the machine and users towards solving the dialog task, as a measure of task success rate.

Unlike task-based systems, non-task-based conversational agents (also known as chitchat models) do not have a specific task to accomplish (e.g. booking a restaurant). The goal of these can arguably be defined as the conversation itself, i.e. the entertainment of the human it is conversing with. Thus, human judgment is still the most reliable evaluation tool we have for such conversational agents. Collecting user ratings for a system, however, is expensive and time-consuming.

In order to deal with these issues, researchers have been investigating automatic metrics for non-task based dialogue evaluation. The most popular of these metrics (e.g. BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005)) rely on surface text similarity (word overlaps) between machine and reference responses to the same utterances. Notwithstanding their popularity, such metrics are hardly compatible with the nature of human dialogue, since there could be multiple appropriate responses to the same utterance with no word overlap. Moreover, these metrics correlate weakly with human judgments (Liu et al., 2016).

Recently, a few studies proposed metrics having a better correlation with human judgment. ADEM (Lowe et al., 2017) is a model trained on appropriateness scores manually annotated at the response-level. Venkatesh et al. (2017) and Guo et al. (2017) combine multiple metrics, each capturing a different aspect of the interaction, and predict conversation-level ratings. In particular, Venkatesh et al. (2017) shows the importance of metrics such as coherence, conversational depth and topic diversity, while Guo et al. (2017) proposes topic-based metrics. However, these studies require extensive manual annotation on top of conversation-level ratings.

In this work, we investigate non-task based dialogue evaluation models trained without relying on any further annotations besides conversation-level user ratings. Our goal is twofold: investigating conversation features which characterize good interactions with a conversational agent and exploring the feasibility of training a model able to predict user ratings in such context.

In order to do so, we utilize a dataset of non-task based spoken conversations between Amazon Alexa users and Roving Mind (Cervone et al., 2017), our open-domain system for the Amazon Alexa Prize Challenge 2017 (Ram et al., 2017). As an upper bound for the rating prediction task, we re-annotate a sample of the corpus using experts and analyse the correlation between expert and user ratings. Afterwards, we analyse the entire corpus using well-known automatically extractable features (user sentiment, Dialogue Acts (both user and machine), conversation length and average user turn length), which show a low, but still significant correlation with user ratings. We show how different combinations of these features together with a LSA representation of the user turns can be used to train a regression model whose predictions also yield a low, but significant correlation with user ratings. Our results indicate the difficulty of predicting how users might rate interactions with a conversational agent.

2 Data Collection

The dataset analysed in this paper was collected over a period of 27 days during the Alexa Prize 2017 semifinals and consists of conversations between our system Roving Mind and Amazon Alexa users of the United States. The users could end the conversation whenever they wanted, using a command. At the end of the interaction users were asked to rate a conversation on a 1 (not satisfied at all) to 5 (very satisfied) Likert scale. Out of all the rated conversations, we selected the ones longer than 3 turns to yield 4,967 conversations. Figure 1 shows the distribution (in percentages) of the ratings in our dataset. The large majority of conversations are between a system and a "first-time" users, as only 5.25% of users had more than one conversation.

3 Methodology

In this section we describe conversation representation features, experimentation, and evaluation methodologies used in the paper.

3.1 Conversation Representation Features

Since in the competition the objective of the system was to entertain users, we expect the ratings to reflect how much they have enjoyed the interaction. User "enjoyment" can be approximated

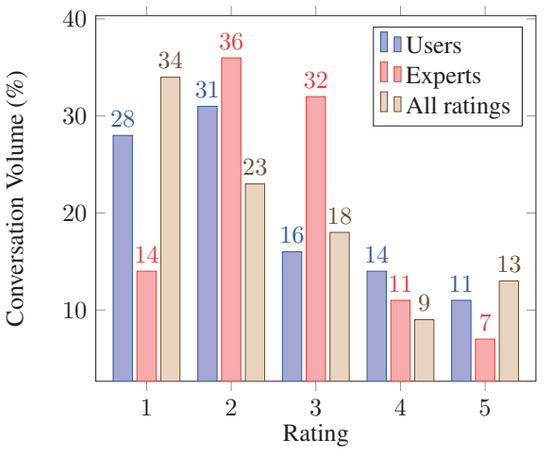


Figure 1: Distribution of user and expert ratings on the annotated random sample of 100 conversations (test set) compared to the distribution of ratings in the entire dataset ("All ratings"). For clarity of presentation, from the latter we excluded the small portion of non integer ratings (2.3% of the dataset).

using different metrics that do not require manual annotation, such as conversation length (in turns), mean turn length (in words), assuming that the more users enjoy the conversation the longer they talk; sentiment polarity – hypothesizing that enjoyable conversations should carry a more positive sentiment. While length metrics are straightforward to compute, the sentiment score is computed using a lexicon-based approach (Kennedy and Inkpen, 2006).

Another representation that could shed a light on enjoyable conversations is Dialogue Acts (DA) of user and machine utterances. DAs are frequently used as a generic representation of intents and the considered labels often include *thanking*, *apologies*, *opinions*, *statements* and alike. Relative frequencies of these tags potentially can be useful to distinguish good and bad conversations. The DA tagger we use is the one described in Mezza et al. (2018) trained on the Switchboard Dialogue Acts corpus (Stolcke et al., 2000), a subset of Switchboard (Godfrey et al., 1992) annotated with DAs (42 categories), using Support Vector Machines. The user and machine DAs are considered as separate vectors and assessed both individually and jointly.

Additional to Dialogue Acts, sentiment and length features, we experiment with word-based text representation. Latent Semantic Analysis (LSA) is used to convert a conversation to a vector. First, we construct a word-document cooccurrence matrix and normalize it. Then, we reduce the dimensionality to 100 by applying Singular Value Decomposition (SVD).

3.2 Correlation Analysis Methodology

The two widely used correlation metrics are Pearson correlation coefficient (PCC) and Spearman's rank correlation coefficient (SRCC). While the former evaluates the linear relationship between variables, the latter evaluates the monotonic one.

The metrics are used to assess correlations of different conversation features, such as sentiment score or conversation length, with the provided human ratings for those conversations; as well as to assess the correlation of the predicted scores of the regression models to those ratings. For the assessment of the correlation of both features and regression models raw rating predictions are used.

3.3 Prediction Methodology

Using the conversation features described above, we train regression models to predict human ratings. We experiment with both Linear Regression and Support Vector Regression (SVR) with radial basis function (RBF) kernel using scikit-learn (Pedregosa et al., 2011). Since the latter consistently outperforms the former, we report only the results for the SVR. The performance of the regression models is evaluated using the standard metrics of Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). Additionally, we compute Pearson and Spearman's Rank Correlation Coefficients for the predictions with respect to the reference human ratings.

We experiment with the 10-fold cross-validation setting. The performance of the regression models is compared to two baselines: (1) mean baseline, where all instances in the testing fold are assigned as a score the mean of the training set ratings, and (2) chance baseline, where an instance is randomly assigned a rating from 1 to 5 with respect to their distribution in the training set. The models are compared for statistical significance to these baselines using paired two-tail T-test with $p < 0.05$. In Section 6 we report average RMSE and MAE as well as average correlation coefficients.

	RMSE	MAE	PCC	SRCC
Exp 1 vs. Exp 2	0.875	0.660	0.705	0.694
Exp 1 vs. Users	1.225	0.966	0.538	0.526
Exp 2 vs. Users	1.286	1.016	0.401	0.370

Table 1: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Pearson (PCC) and Spearman's rank (SRCC) correlation coefficients among user and expert ratings.

4 Upper bound

Since human ratings are inherently subjective, and different users can rate the same conversation differently, it is difficult to expect the models to yield perfect correlations or very low RMSE and MAE. In order to test this hypothesis two human experts (members of our Alexa Prize team) were asked to rate a random subset of the corpus (100 conversations). The rating distributions for both experts and users on the sample is reported in Figure 1. We observe that expert ratings tend to be closer to the middle of the Likert scale (i.e. from 2 to 4), while users had more conversations with ratings at both extremes of the scale (i.e. 1 and 5).

The RMSE, MAE and Pearson and Spearman's rank correlation coefficients of expert and user ratings are reported in Table 1. We observe that the experts tend to agree with each other more than they agree individually with users, since compared to each other the experts have the highest Pearson and Spearman correlation scores (0.705 and 0.694, respectively) and the lowest RMSE and MAE (0.875 and 0.660, respectively). The fact that expert ratings do not correlate with user ratings as well as they correlate among themselves, confirms the difficulty of the task of predicting subjective user ratings even for humans.

5 Correlation Analysis Results

The results of the correlation analysis are reported in Table 2. From the table, we can observe that conversation length has a positive correlation with human judgment, while the average user turn length has a negative correlation. The positive correlation with conversation length confirms the expectation that users tend to have longer conversations with the system when they enjoy it. The negative correlation with average user turn length, on the other hand, is unexpected. As expected, sentiment score has a significant positive correlation with human judgments.

Feature	PCC	SRCC
Conversation Length	0.133**	0.111**
Av. User Turn Length	-0.068**	-0.079**
User Sentiment	0.071**	0.088**
User Dialogue Acts		
yes-answer	0.081**	0.088**
appreciation	0.070**	0.115**
thanking	0.062**	0.089**
action-directive	-0.069**	-0.052**
statement-non-opinion	0.050**	0.037*
...		
Machine Dialogue Acts		
yes-no-question	0.042**	0.038**
statement-opinion	-0.027*	-0.032*
...		

Table 2: Pearson (PCC) and Spearman's rank (SRCC) correlation coefficients for conversation lengths, sentiment score, and user and machine Dialogue Acts. Correlations significant with $p < 0.05$ are marked with * and $p < 0.01$ with **.

Due to the space considerations, we report only a portion of the DAs that have significant correlations with human ratings. The analysis confirms our expectations that user DAs, such as *thanking* and *appreciation*, have significant positive correlations. We also observe that the *action-directive* DA has a negative correlation. Since this DA label covers the turns where a user issues control commands to the system, we hypothesize this correlation could be due to the fact that in such cases users were using a task-based approach with our system which was instead designed for chitchat and might therefore feel disappointed (e.g. requesting the Roving Mind system to perform actions it was not designed to perform, such as playing music).

Regarding machine DAs, we observe that even though some DAs exhibit significant correlations, overall they are lower than user DAs. In particular, *yes-no-question* has a significant positive correlation with human judgments, indicating that some users appreciate machine initiative in the conversation. The analysis confirms the utility of length and sentiment features, as well as the importance of some DAs (generic intents) for estimating user ratings.

6 Prediction Results

The results of the experiments using 10-fold cross-validation and Support Vector Regression are reported in Table 3. We report performances of each feature representation is isolation and their combi-

	RMSE	MAE	PCC	SRCC
BL: Chance	1.967	1.535	0.007	0.023
BL: Mean	1.382	1.189	N/A	N/A
Lengths	1.400	1.116*	0.153*	0.158**
Sentiment	1.423	1.128*	0.109*	0.122*
DA: user	1.378	1.106*	0.213**	0,207**
DA: machine	1.418	1.129*	0.104*	0.099*
DA: user+machine	1.375	1.106*	0.219**	0.211**
LSA	**1.350***	**1.075***	0.299**	0.288**
All - LSA	1.366*	1.100*	0.240**	0.230**
All	**1.350***	1.078*	**0.303****	**0.290****

Table 3: 10 fold cross-validation average Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Pearson (PCC) and Spearman's rank (SRCC) correlation coefficients for regression models. RMSE and MAE significantly better than the baselines are marked with *. Correlations significant with $p < 0.05$ are marked with * and $p < 0.01$ with **.

nations. We consider two baselines – chance and mean. For the chance baseline an instance is randomly assigned a rating with respect to the training set distribution. For the mean baseline, on the other hand, all the instances are assigned the mean of the training set as a rating. The mean baseline yields better RMSE and MAE scores; consequently, we compare the regression models to it.

Sentiment and length features (conversation and average user turn) both yield RMSE higher than the mean baseline and MAE significantly lower than it. Nonetheless, their predictions have significant positive correlations with reference human ratings. The picture is similar for the models trained on user and machine DAs alone and their combination. The RMSE scores are higher or insignificantly lower and MAE scores are significantly lower than the mean baseline.

For the LSA representation of conversations we consider ngram sizes between 1 and 4. The representation that considers 4-grams and the SVD dimension of 100 yields better performances; thus, we report the performances of this models only, and use it for feature combination experiments. The LSA model yields significantly lower error both in terms of RMSE and MAE. Additionally, the correlation of the predictions is higher than for the other features (and combinations).

The regression model trained on all features but LSA, yields performances significantly better than the mean baseline. However, they are inferior to that of LSA alone. Combination of all the features retains the best RMSE of the LSA model, but achieves a little worse MAE score. While it yields the best Pearson and Spearman's rank correlation coefficients among all the models, the difference from LSA only model is not statistically relevant using Fisher r-to-z transformation.

7 Conclusions

In this work we experimented with a set of automatically extractable black-box features which correlate with the human perception of the quality of interactions with a conversational agent. Furthermore, we showed how these features can be combined to train automatic non-task-based dialogue evaluation models which correlate with human judgments without further expensive annotations.

The results of our experiments and analysis contribute to the body of observations that indicate that there still remains a lot of research to be done in order to understand characteristics of enjoyable conversations with open-domain non-task oriented agents. In particular, our analysis of expert vs. user ratings suggests that the task of estimating subjective user ratings is a difficult one, since the same conversation might be rated quite differently.

For the future work, we plan to extend our corpus to include interactions with multiple conversational agents and task-based systems, as well as to explore other features that might be relevant for assessing human judgment of interaction with a conversational agent (e.g. emotion recognition).

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72.

Alessandra Cervone, Giuliano Tortoreto, Stefano Mezza, Enrico Gambi, and Giuseppe Riccardi. 2017. Roving mind: a balancing act between open–domain and engaging dialogue systems. In *Alexa Prize Proceedings*.

Morena Danieli and Elisabetta Gerbino. 1995. Metrics for evaluating dialogue strategies in a spoken language system. In *Proceedings of the 1995 AAAI spring symposium on Empirical Methods in Discourse Interpretation and Generation*, volume 16, pages 34–39.

John J Godfrey, Edward C Holliman, and Jane McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 1, pages 517–520. IEEE.

Fenfei Guo, Angeliki Metallinou, Chandra Khatri, Anirudh Raju, Anu Venkatesh, and Ashwin Ram. 2017. Topic-based evaluation for conversational bots. In *NIPS 2017 Conversational AI workshop*.

Alistair Kennedy and Diana Inkpen. 2006. Sentiment classification of movie reviews using contextual valence shifters. *Computational intelligence*, 22(2):110–125.

Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132.

Ryan Lowe, Michael Noseworthy, Iulian Vlad Serban, Nicolas Angelard-Gontier, Yoshua Bengio, and Joelle Pineau. 2017. Towards an automatic turing test: Learning to evaluate dialogue responses. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1116–1126.

Stefano Mezza, Alessandra Cervone, Giuliano Tortoreto, Evgeny A. Stepanov, and Giuseppe Riccardi. 2018. Iso-standard domain-independent dialogue act tagging for conversational agents. In *Proceedings of COLING 2018, the 27th International Conference on Computational Linguistics: Technical Papers*, pages 3539–3551.

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 on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*.

Ashwin Ram, Rohit Prasad, Chandra Khatri, Anu Venkatesh, Raefer Gabriel, Qing Liu, Jeff Nunn, Behnam Hedayatnia, Ming Cheng, Ashish Nagar, Eric King, Kate Bland, Amanda Wartick, Yi Pan, Han Song, Sk Jayadevan, Gene Hwang, and Art Pettigrue. 2017. Conversational ai: The science behind the alexa prize. In *Alexa Prize Proceedings*.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational linguistics*, 26(3):339–373.

Anu Venkatesh, Chandra Khatri, Ashwin Ram, Fenfei Guo, Raefer Gabriel, Ashish Nagar, Rohit Prasad, Ming Cheng, Benham Hedayatnia, Angeliki Metallinou, Rahul Goel, Shaohua Yang, and Anirudh Raju. 2017. On evaluating and comparing conversational agents. In *NIPS 2017 Conversational AI workshop*.

Marilyn A Walker, Diane J Litman, Candace A Kamm, and Alicia Abella. 1997. Paradise: A framework for evaluating spoken dialogue agents. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 271–280. Association for Computational Linguistics.

An efficient Trie for binding (and movement)

Cristiano Chesi
NETS - IUSS
P.zza Vittoria 15
I-27100 Pavia (Italy)
cristiano.chesi@iusspavia.it

Abstract

English. Non-local dependencies connecting distant structural chunks are often modeled using (LIFO) memory buffers (see Chesi 2012 for a review). Other solutions (e.g. *slash* features in HPSG, Pollard & Sag 1994) are not directly usable both in parsing and in generation algorithms without undermining an incremental left-right processing assumption. Memory buffers are however empirically limited and psycholinguistically invalid (Nairne 2002). Here I propose to adopt *Trie* memories instead of *stacks*. This leads to simpler and more transparent solutions for establishing non-local dependencies both for *wh-* argumental configurations and for anaphoric pronominal coreference.

Italian. *Nell'implementazione di dipendenze non locali che mettano in connessione due costituenti arbitrariamente distanti in una struttura frasale, spesso si è ricorsi all'uso di memorie a pila (LIFO; si veda Chesi 2012 per una panoramica sul tema). Le altre soluzioni proposte (e.g. tratti* slash *in HPSG, Pollard & Sag 1994) non risultano implementabili in modo trasparente, né in generazione né in parsing, con algoritmi che tengano conto del requisito di incrementalità del processamento. Tuttavia, viste le limitazioni psicolinguistiche ed empiriche delle memorie a pila (Nairne 2002), qui si propone di adottare memorie di tipo* Trie *per codificare i tratti rilevanti nello stabilire dipendenze non locali nel caso di strutture che impiegano elementi wh- argomentali e nel legamento pronominale anaforico.*

1 Introduction

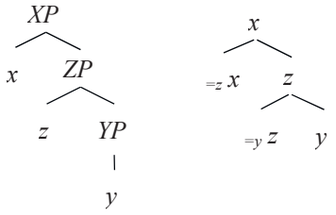
Relations among structural chunks in a sentence are not always resolvable using strictly local dependencies. This is the case of argumental *wh-*items in languages like English (or Italian), where the argument and the predicate can be arbitrarily distant, (1).a. Another case of non-local dependency is pronominal coreference that in some cases can also be cross-sentential, (1).b-b', (1).b-b".

(1) a. [$_X$ Cosa] (tu) pensi che (io) [$_Y$ mangi_]?
what (you) think that (I) eat$_{SUBJ\text{-}1P\text{-}Sing}$
what do you think I eat?
b. [$_X$ Gianni]$_i$ saluta [$_Z$ Mario]$_j$.
G. *says hello* *(to) M.*
b'. Poi *pro* $_i$ [$_Y$ si]$_i$ lava.
then (he) himself$_j$ washes.
then he washes himself
b". Poi *pro* $_i$ [$_Y$ lo]$_j$ lava.
then (he) him$_j$ washes.
then he washes him

From a purely structural perspective, the chunks *X* and *Y* enter a non-local dependency relation when some material *Z* intervenes between them. A long tradition of different approaches addressed this issue from different perspective (see Nivre 2008, for instance, for a comparison among Stack-based and List-based algorithms in parsing). Most of the time these approaches rely on transformations of the grammar into a deductive system for both parsing (Shieber et al. 1995) and generation (Shieber 1988). A loss of transparency with respect to the linguistic intuitions that motivated a specific grammatical formalism is then at issue. Here I will argue in favor of a simple derivational and deterministic perspective in which phrases are considered the result of the recursive application of structure building operations (Chomsky 1995). In its simplest format, classic structural descriptions, (2).a, reduce to lexicalized

trees, (2).a', in which *x* and *z* creates a constituent (get *merged*) either if *x selects z* (=z x, in Stabler's 1997 formalism) or the way around (=x z). Leaves are linearly ordered and constituents labels reduce to the selecting lexical items.

(2) a. a'.



By definition, *x* and *y* cannot enter a local dependency whenever an intervening item *z* blocks a local selection between *x* and *y*. There are cases, however, in which *x* and *y* should enter a local selection relation: in (1).a, *x* receives a thematic role from *y*, hence *y* should select *x* according to the uniformity of theta-role assignment hypothesis (Baker 1988). In this case, a non-local dependency must be established. Implementing the *movement* metaphor (Stabler 1997) in top-down terms, Chesi (2017) proposes that an item *x* is *moved* into a Last-In-First-Out (LIFO) memory buffer (M) whenever it brings into the computation features that are unselected: if a (categorial) feature $_X$ is selected and a lexical item *a* brings $_X$ but also $_Y$ from the lexicon (i.e. [$_X$ $_Y$ *a*]]), then *a* gets merged (i.e. [$\underline{x}$[$\underline{x}$ $_Y$ *a*]]), but the unselected feature [$_Y$ *(a)*] is *moved* into the last position (the most prominent one) of the M-buffer. As soon as a feature $_Y$ will be selected (=Y), the last item in the memory buffer, if bearing the relevant $_Y$ category, will be remerged in the structure before any other item from the lexicon, the satisfying a local selection requirement. After its re-merge, the item is removed from the M-buffer.

This paper proposes a theoretical solution for simplifying this memory-based approach without losing any descriptive adequacy: here I will do away with the buffer idea (and, as a consequence, with the LIFO restrictions) by postulating a memory Trie (Fredkin 1960) based on the features merged in the structure during the derivation. I will show that this solution is psycholinguistically more plausible than LIFO buffers used so far and computationally sound.

1.1 Implementing non-local dependencies

Phase-based Minimalist Grammars (PMG, Chesi 2007) express top-down, left-right derivations that can be used directly both in generation and in parsing (Chesi 2012, see Chesi 2017 for some advantages for predicting difficulty in parsing). Non-local dependencies of the (1).a kind are established whenever a constituent lexicalizes an expected feature but also brings into the structure unexpected features that should be selected later on, in order for the sentence to be grammatical. This is implemented using PMGs able to deal with non-local dependencies as discussed below.

1.1.1 A simpler PMG formalization

PMGs are lexicalized grammars in which structure building operations are included in the grammatical formalism (Chesi 2007 and Collins & Stabler 2016 for a recent formalization of MGs). Unlike other formalisms (e.g. CFGs, HPSGs, TAGs or CCGs) PMGs do not simply express a declarative knowledge but also a deterministic procedure (Marcus 1980, Shieber 1983) that explicitly produces, step-by-step, a full derivation which should be common both in parsing and in generation (Momma & Phillips 2018). Below the basic definitions representing a simplified formalization of the crucial components of a PMG: categories, feature structures, lexical items, structure building operations and their triggers.

Definition 1 *A* category *is a morpho-syntactic feature with a(n optional) value specification:* [$_{cat(:value)}$]. *Each derivation starts with a (default) projection of a specific category (*phase edge*).*

Even if this is not strictly necessary here, for simplicity, categories will be divided into *functional* (e.g. [$_{D:definite}$] or simply [$_D$] for a definite determiners/articles), *phase edges* (functional categories introducing a new phase, in the sense of Chomsky 2008), and *lexical* (e.g. nominal or verbal categories, namely the sole categories, a part from the default root selection that starts the derivation, entitled to select new phase edges).

Definition 2 *A* lexical item *is a ordered feature structure (Attribute-Value Matrix) encoding* phonetic *(/phon),* semantic *(#sem) and* category *features:* [$_{cat\_1(:v\_1)\ \ldots\ cat\_n(:v\_n)}$ *#sem /phon*]

Neither phonetic (instruction for pronouncing a lexical item) nor semantic features (instruction for interpreting the item both lexically, e.g. WordNet synset, Miller 1995, and compositionally, e.g. specification of a functional application, Heim & Kratzer 1998) will be discussed here. I will use simpler entries like [$_N$ man] (by default: *num:sg*, *gen:male*). Certain items might be optionally specified for some categories: [$_{(F)\ X}$ ...] indicates that the $_F$ category (*focus*) can be present or not (this has semantic and a derivational impact).

Definition 3 *A* phrase structure *is a hierarchical feature structure combining categories and lexical items; a phrase structure is* fully lexicalized *iff each* category *in it is associated to a* lexical item*.*

Definition 4 *An* edge category *is the most prominent feature, namely the target of any* structure building operation*;*

By default, edge categories (that will be *underlined* below) are the left-most feature of any lexical item and the right-most feature of any unlexicalized phrase structure. If an optional category is present, this is the edge of the lexical item.

Definition 5 Structure building operations *are functions taking in input phrase structures and returning modified phrase structures.* Merge*,* Move *and* Expect *are structure building operations.*

Definition 6 Merge *is a binary* structure building operation *that unifies the edge categories in a* phrase structure *and a* lexical item*:*
Merge([X … [Y]], [Y … lex]) → [X … [¥[¥ … lex]]]

Definition 7 Expect *takes as input a select feature and introduce it in the structure:* [=X] → [=X [X]]

An *expectation/expansion* is then a lexically or categorically encoded select feature; whenever *categories* in the lexicon are specified for select features (e.g. [X =Z]), those select features must be expanded after lexicalization (i.e. first *merge*: [¥[X …] =Z], then *expect*: [¥[X …] =Z[Z]])

Definition 8 *An* unexpected category *is any* unselected feature *introduced in the derivation by merging a lexical item bearing both the expected feature(s) and unexpected one(s).*

e.g. merge([... [Y]], [Y Z … *a*]) → [... [¥[¥ Z *a*]]] *Unselected item after* merge: [¥ Z *(a)*]

Definition 9 Move *is the operation storing items with* unexpected features *in a LIFO M(emory)-buffer*. [... [¥[¥ Z *a*]]] → M:<[¥ Z *(a)*]>

Since the lexical items is already pronounced, phonetic features will not be re-merged, hence *(a)*.

Definition 10 M-buffer *must be empty at the end of the derivation.* Lexical items *stored in the memory buffer must be (re-)merged, as soon as a compatible* expectation *is introduced, before any other* lexical item*.*

1.1.2 A toy grammar exemplifying processing of non-local dependencies

Given the (simplified) lexicon in (3), the generation of (1).a proceeds as indicated in (4):

(3) simplified lexicon for generating and parsing sentences in (1):

Lexicon
[(S) D N anim G./M.], [F D gen:fem N cosa], [D:reflex si$_x$], [(S) D pers:1 case:nom N (io)], [(S) D pers:2 case:nom N (tu)], [C che], [C poi], [Pers:1 T V mangi =D:case:nom =D:case:acc], [pers:2 T V pensi =D:case:nom =C], [pers:3 T V lava =D:reflex:anim =D:case:acc]
Categories
Phase edges (functional categories): [C =S], [F =S], [D =N] *Other functional categories*: [S =T], [T =V] *Lexical categories*: [N], [V]

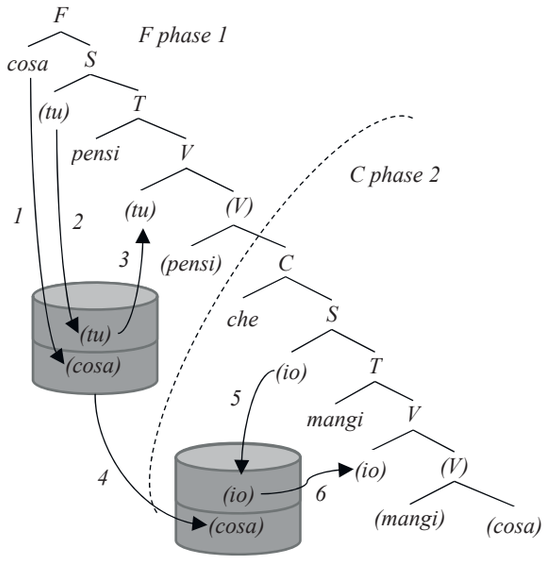
(4) Generation of (1).a
Cosa$_i$ (tu) pensi che (io) mangi _$_i$?

1. [F =S] *(default root phase edge expectation)*
2. [¥[F D … cosa] =S] *(merge)*
3. [¥[F D … cosa] =S] M<[D … (cosa)]> *(move)*
4. [¥[F D … cosa] =S[S =T]] *(expect)*
5. [¥[F D … cosa] =S[S[S D … (tu)] =T]] *(merge)*
6. [¥[F D … cosa] =S[S[S D … (tu)] =T]] *(move)*
   M<[D … (cosa)], [D … (tu)] >
7. [¥[F D … cosa] =S[S[S D … (tu)] =T[T =V]]] *(expect)*
8. [¥[F D … cosa] =S[S[S D … (tu)] =T[T =V[T V pensi =D =C]]]] *(merge)*
9. … [... pensi =D[D =N] =C] *(expect)*
10. … [... pensi =D[D =N [D … (tu)]] =C] *(merge from M)*
11. … =C[C =S]] *(expect)*
12. … =C[C[C che] =S]] *(merge)*
13. … =C[C[C che] =S[S =T]]] *(expect)*
14. … =C[C[C che] =S[S[S D… (io)] =T]]] *(merge)*
15. … =C[C[C che] =S[S[S D… (io)] =T]]] *(move)*
    M<[D … (cosa)], [D … (io)] >
16. … =C[C[C che] =S[S[S D… (io)] =T[T =V]]]] *(expect)*
17. … [T =V [... T V mangi =D =D]] *(merge)*
18. [... mangi =D[D =N] =D]] *(expect)*
19. [... mangi =D[D =N [D … (io)]] =D]] *(merge from M)*
20. [... mangi =D[D =N [D … (io)]] =D[D =N]]] *(expect)*
21. [... mangi =D[D =N [D … (io)]] =D[D =N [D … (cosa)]]]] *(merge from M)*

The sentence is grammatical iff the M-buffer is emptied by the end of the derivation and no expectations are pending. The structural description (to be considered as the history of the derivation, which is also a representation of all the useful structural restrictions) is represented in (5). The features triggering *Merge*, *Move* and *Expect* are omitted in the tree for simplicity (refer to (3) and (4) for the full set of features and for the step by step derivation). Notice that "vacuous" movements of the null subjects in Italian is the main difference between generation and parsing: in parsing, an underspecified (for number and person) null subject is postulated then re-merged (unified with the relevant feature values) after the verbal morphology has been analyzed. Moreover, using the toy grammar in (3), 3 expectations could

initialize the parsing (*C*, *F* and *D*), but only the first one (*F*) would result compatible with the "cosa pensi" incipit of the sentence (cf. Earley 1977).

(5) Tree diagram summarizing the step-by-step derivation in (4)



1.2 Non-local pronominal coreference

The same strategy cannot be used for pronominal binding, e.g. (1).b-b', since:

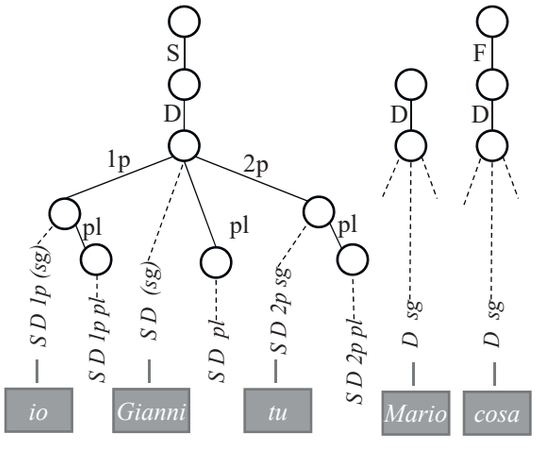
i. LIFO memory buffers are populated only for a short amount of time, then got emptied as soon as the relevant features are selected; referential items should stay in memory longer after the item has been selected for capturing also (cross-sentential) binding effects.
ii. LIFO structure is not suitable to capture crossing dependencies like the one in (1).b-b'.

Problem i. has been discussed and resolved both by Schlenker (2005) and Bianchi (2009) by postulating "referential buffers" of the kind we discussed in §1.1 in which referential NPs are stored and used without being removed for binding (i.e. coindexing) in anaphoric items. Bianchi (2009) shows how local and global referential buffers are sufficient to capture violation of binding principles: local buffers are phase-specific, hence nested phase buffers are inaccessible from higher phase-buffers, higher phase-buffers are accessible from lower phases, while a global referential buffer is accessible by all phases. With this distinction, Principle C effects (rephrasing Chomsky 1981, a pronoun cannot be co-referent with a non-pronominal that it c-commands: "He said that Bill is funny". He ≠ Bill) is the result of the application of a non-redundancy principle, favoring the usage of a anaphor instead of a referential expression that would re-insert a referential item already present in the referential buffer. Bianchi (2009) also notices that for retrieving the correct referent from a referential buffer we need to depart from the LIFO structure assumed so far.

2 Trie memories for capturing non-local dependencies

One way to implement Bianchi's idea (§1.2) in an efficient way is to use *Trie* memories. Tries (from re*trie*val), in their simplest form, are hierarchical, acyclic data structures that guarantee fast insertion, search and deletion of information (Fredkin 1960). Tries are often used in parsing for efficient encoding of phrase structures (Leermakers 1992 and Moore 2000 a.o.). Indeed, more efficient formats for representing, for instance, CFG phrase rules exist: *Minimized FSAs*, compared to *Tries*, perform generally better (Klein & Manning 2001). Here I will argue that, despite their lower performance compared to other phrase structure transformations, they better support correct empirical predictions both in case of coreferential binding and *wh-* movement, so they are worth to be considered both for empirical and psycholinguistic reasons. The original part of this proposal is related to the storage, in Tries format, of referential features encoded in the phrase structure built so far as indicated below (root node omitted):

(6) Trie memory fragment



Each referential NP is identified by a specific path starting from the root and reaching one leaf of the common Trie representing in a compact way all the relevant features related to any referential item inserted in the derivation. If "you" is merged in the structure as a subject, its root would be the "S"

(topic) feature; “cosa” would be identified by the path F-D (3rd person being the default person, or no person, Sigurdsson 2004 and singular the default number); “io” would be S-D-1p, “tu” S-D-2p, “Gianni” S-D and “Mario” simply D (other irrelevant features being omitted for clarity). Few interesting facts are worth highlighting here:

1. Two NPs will be distinct if and only if a distinct path identifies them: with such a feature structure, “cosa” and “casa” would be undistinguishable; for separating the two, extra features must be added to the Trie (e.g. *animacy*);
2. The more similar a path, the faster the insertion in memory would be, but the easier it would also be to confound them at retrieval: storing “tu” after “voi” would be faster than storing “io” after “tu”; similarly, confounding “tu” with “voi” is expected to be easier than confounding “tu” with “io”, though the number of features stored is the same;

It is clear that the fragment in (6) must be expanded including “semantic” features like *animacy*, *mass*/*countable* etc. that can be selected by the relevant predicate then creating distinct paths. Nevertheless, these two facts are already sufficient to subsume the similarity effects discussed in Chesi (2017) without relying to memory stacks.

2.1 Capturing pronominal coreference

An anaphoric item, for receiving its correct co-referent binding index, triggers an inspection of the features that qualify the items in memory as good binders, namely *topics* matching *person*, *number* and *gender* features. In (1).b-b' and (1).b-b" a (third person, in this case) null subject is (always) used anaphorically in Italian, then, in order to be correctly interpreted it must be co-referent with a 3rd person, animate, singular, male binder. This would be only compatible with “Gianni” which is first merged in a topic (S) position and it has all the relevant features. Even though “G” shares any other feature with the direct object “Mario”, its topic insertion position is crucial from selecting G instead of M. The Trie idea then supports the correct retrieval forcing distinct traversal starting with the highest feature encoded. This is much more efficient than revisiting LIFO assumptions. Notice also that this does not overgenerate: according to the binding principles, an anaphor “si” and not a “pronoun”, should be co-indexed in its “local” domain. This is obtained by letting “si” look for the topic encoded feature while “lo” would inspect only compatible, non-locally topicalized, items (e.g. “M” in (1).b-b").

2.2 Capturing movement in general

While referents in this Trie are not removed once an item is retrieved (but possibly receive a boost in its accessibility, Lewis & Vasishth 2005), a movement-based dependencies need to remove the relevant item after remerge. Here I propose to use the very same Trie representation, (6), and mark the “unexpected” features identifying an unselected item. Remember that in order to remerge the correct item, the features cued by the selecting head must be selected and a distinct path should be found in the Trie: steps 10 and 19 in (4) require a specific set of features to be retrieved that in the Trie correspond to the path D-2p and D-1p respectively. This path identifies uniquely the item “tu” and “io”, while another item (“cosa”, D-sg) is stored in memory. Without need of a LIFO structure we can then retrieve effectively the correct item without confusion, then removing the “unexpected” marks from the features for the unique path identifying the remerged item just retrieved.

3 Conclusion

In this paper, I presented a revision of the memory buffer used for parsing and generation in PMGs: instead of using a classic LIFO memory, proved to be sufficient to capture locality effects (Friedmann et al. 2009) when “similar” NPs are processed (Warren & Gibson 2005, Chesi 2017), but not fully plausible from a psycholinguistic perspective (no serial order seems to be relevant at retrieval, Nairne 2002, as we saw also in case of pronominal binding), I defined a Trie memory replacement, based on feature hierarchies sensitive to the structural insertion point of the memorized item. This prevents order of insertion from being strictly relevant at retrieval, without losing any ability to discriminate the correct items to be recalled for establishing a relevant (non-local) structural dependency both in thematic role assignment or anaphoric binding contexts. The Trie structure here proposed is clearly a bit simplistic, though based on a relevant evidence suggesting that person features are “higher” in the structure than “number” features (Mancini et al. 2011). Other (semantic) features should be included (e.g. animacy) as well as prosodic/salience markers (Topic, New Information/Contrastive Focus, Kiss 1998) that clearly play a role in making salient (i.e. unique in a Trie) a specific item, possibly relating the “fluctuation” of prominence of items stored in memory (Lewis & Vasishth 2005) to precise structural proprieties.

Reference

Baker, M. C. 1988. *Incorporation: A theory of grammatical function changing*. Chicago: University of Chicago Press

Bianchi, V. 2009. A note on backward anaphora. *Rivista di Grammatica Generativa*, 34, 3-34.

Chesi C. 2017. Phase-based Minimalist Parsing and complexity in non-local dependencies. *Proceedings of CLiC-it 2017*. CEUR workshop proceedings, ROMA:CEUR. Rome, 11-13 Dec 2017, doi: urn:nbn:de:0074-2006-4

Chesi, C. 2007. An introduction to Phase-based Minimalist Grammars: why move is Top-Down from Left-to-Right. *Studies in Linguistics*, 1, 49-90.

Chesi, C. 2012. *Competence and Computation: toward a processing friendly minimalist Grammar*. Padova: Unipress.

Chomsky, N. 1981. *Lectures on government and binding: The Pisa lectures*. Berlin: Walter de Gruyter.

Chomsky, N. 1995. *The Minimalist Program*. Cambridge (MA): MIT Press.

Chomsky, N. 2008. On phases. In C. Freidin, P. Otero, M. L. Zubizarreta (eds.) *Foundational issues in linguistic theory: Essays in honor of Jean-Roger Vergnaud*. MIT Press.

Collins, C., & E. Stabler. 2016. A formalization of minimalist syntax. *Syntax*, 19, 1: 43-78.

Earley, J. 1970. An efficient context-free parsing algorithm. *Communications of the Association for Computing Machinery*, 13(2), February.

Fredkin, E., 1960. Trie memory. *Communications of the ACM*, 3(9), pp.490-499.

Friedmann, N., Belletti, A., & Rizzi, L. 2009. Relativized relatives: Types of intervention in the acquisition of A-bar dependencies. *Lingua*, 119(1), 67-88.

Heim I. & A. Kratzer. 1998. *Semantics in generative grammar*. Oxford: Blackwell.

Kiss, K. É. 1998. Identificational focus versus information focus. *Language*, 74(2), 245-273.

Klein, D., & Manning, C. D. 2001. Parsing with treebank grammars: Empirical bounds, theoretical models, and the structure of the Penn treebank. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics* (pp. 338-345). Association for Computational Linguistics.

Leermakers, R. 1992. A recursive ascent Earley parser. *Information Processing Letters*, 41:87–91.

Lewis, R. L., & Vasishth, S. 2005. An activation-based model of sentence processing as skilled memory retrieval. *Cognitive science*, 29(3), 375-419.

Mancini, S., Molinaro, N., Rizzi, L., & Carreiras, M. 2011. A person is not a number: Discourse involvement in subject–verb agreement computation. *Brain research*, 1410, 64-76.

Marcus, M. P. 1980. *Theory of syntactic recognition for natural languages*. MIT press.

Miller, G. A. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11), 39-41.

Momma, S., & Phillips, C. (2018). The relationship between parsing and generation. *Annual Review of Linguistics*, 4, 233-254.

Moore R. C. 2000. Improved left-corner chart parsing for large context-free grammars. In *Proceedings of the Sixth International Workshop on Parsing Technologies*.

Nairne, J. S. 2002. The myth of the encoding-retrieval match. *Memory*, 10(5-6), 389-395.

Pollard, C. and Sag, I.A., 1994. *Head-driven phrase structure grammar*. University of Chicago Press.

Schlenker, P. 2005. Non-redundancy: Towards a semantic reinterpretation of binding theory. *Natural Language Semantics*, 13(1), 1-92.

Shieber, S. M. 1983. Sentence disambiguation by a shift-reduce parsing technique. In *Proceedings of the 21st annual meeting on Association for Computational Linguistics* (pp. 113-118). Association for Computational Linguistics.

Shieber, S. M. 1988. A uniform architecture for parsing and generation. In *Proceedings of the 12th conference on Computational linguistic*s. Vol. 2 (pp. 614-619). Association for Computational Linguistics.

Shieber, S. M., Schabes, Y., & Pereira, F. C. 1995. Principles and implementation of deductive parsing. The Journal of logic programming, 24(1-2), 3-36.

Sigurdsson, H. A. 2004. The syntax of person, tense and speech features. *Italian Journal of Linguistics*, 16, 219-251.

Stabler, E. 1997. Derivational minimalism. In *International Conference on Logical Aspects of Computational Linguistics* (pp. 68-95). Springer, Berlin, Heidelberg.

Stabler, E. 2013. Two Models of Minimalist, Incremental Syntactic Analysis, *Topics in Cognitive Science*, 5:611–633, doi:10.1111/tops.12031.

Van Dyke, J. A., & McElree, B. 2006. Retrieval interference in sentence comprehension. *Journal of Memory and Language*, 55(2), 157-166.

Warren, T., & Gibson, E. 2005. Effects of NP type in reading cleft sentences in English. *Language and Cognitive Processes*, 20(6), 751-767.

Generalizing Representations of Lexical Semantic Relations

Anupama Chingacham
SFB 1102, Saarland University
Saarbrucken ,66123, Germany
anu.vgopal2009@gmail.com

Denis Paperno
CNRS, LORIA, UMR 7503
Vandoeuvre-lès-Nancy, F-54500, France
denis.paperno@loria.fr

Abstract

English. We propose a new method for unsupervised learning of embeddings for lexical relations in word pairs. The model is trained on predicting the contexts in which a word pair appears together in corpora, then generalized to account for new and unseen word pairs. This allows us to overcome the data sparsity issues inherent in existing relation embedding learning setups without the need to go back to the corpora to collect additional data for new pairs.

Italiano. *Proponiamo un nuovo metodo per l'apprendimento non supervisionato delle rappresentazioni delle relazioni lessicali fra coppie di parole (word pair embeddings). Il modello viene allenato a prevedere i contesti in cui compare uns coppia di parole, e successivamente viene generalizzato a coppie di parole nuove o non attestate. Questo ci consente di superare i problemi dovuti alla scarsità di dati tipica dei sistemi di apprendimento di rappresentazioni, senza la necessità di tornare ai corpora per raccogliere dati per nuove coppie di parole.*

1 Introduction

In this paper we address the problem of unsupervised learning of lexical relations between any two words. We take the approach of unsupervised representation learning from distribution in corpora, as familiar from word embedding methods, and enhance it with an additional technique to overcome data sparsity.

Word embedding models give a promise of learning word meaning from easily available text data in an unsupervised fashion and indeed the resulting vectors contain a lot of information about the semantic properties of words and objects they refer to, cf. for instance Herbelot and Vecchi (2015). Based on the distributional hypothesis coined by Z. S. Harris (1954), word embedding models, which construct word meaning representations as numeric vectors based on the co-occurrence statistics on the word's context, have been gaining ground due to their quality and simplicity. Produced by efficient and robust implementations such as word2vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014), modern word vector models are able to predict whether two words are related in meaning, reaching human performance on benchmarks like WordSim353 (Agirre et al., 2009) and MEN (Bruni et al., 2014).

On the other hand, lexical knowledge includes not only properties of individual words but also relations between words. To some extent, lexical semantic relations can be recovered from the word representations via the vector offset method as evidenced by various applications including analogy solving, but already on this task it has multiple drawbacks (Linzen, 2016) and has a better unsupervised alternative (Levy and Goldberg, 2014).

Just like a word representation is inferred from the contexts in which the word occurs, information about the relation in a given word pair can be extracted from the statistics of contexts in which the two words of the pair appear together. In our model, we use this principle to learn high-quality pair embeddings from frequent noun pairs, and on their basis, build a way to construct a relation representation for an arbitrary pair.

Note that we approach the problem from the viewpoint of lerning general-purpose semantic knowledge. Our goal is to provide a vector representation for an arbitrary pair of words w_1, w_2. This is a more general task than *relation extraction*, which aims at identifying the semantic rela-

tion between the two words in a particular context. Modeling such general relational knowledge is crucial for natural language understanding in realistic settings. It may be especially useful for recovering the notoriously difficult bridging relations in discourse since they involve understanding implicit links between words in the text.

Representations of word relations have applications in many NLP tasks. For example, they could be extremely useful for resolving bridging, especially of the lexical type (Rösiger et al., 2018). But in order to be useful in practice, word relation models must generalize to rare or unseen cases.

2 Related Work

Our project is related to the task of relation extraction that has been in focus of various complex models (Mintz et al., 2009; Zelenko et al., 2003) including recurrent (Takase et al., 2016) and convolutional neural network architectures (Xu et al., 2015; Nguyen and Grishman, 2015; Zeng et al., 2014), although the simple averaging or summation of the context word vectors seems to produce good results for the task (Fan et al., 2015; Hashimoto et al., 2015). The latter work by Hashimoto et al. bears the greatest resemblance to the approach to learning semantic relation representations that we utilize here. Hashimoto et al. train noun embeddings on the task of predicting words occurring in between the two nouns in text corpora and use these embeddings along with averaging-based context embeddings as input to relation classification.

There are numerous studies dedicated to characterizing relations in word pairs abstracted away from the specific context in which the word pair appears. Much of this literature focuses on one specific lexical semantic relation at a time. Among these, lexical entailment (hypernymy) has probably been the most popular since Hearst (1992) with various representation learning approaches specifically targeting lexical entailment (Fu et al., 2014; Anh et al., 2016; Roller and Erk, 2016; Bowman, 2016; Kruszewski et al., 2015) and the antonymy relation has also received considerable attention (Ono et al., 2015; Pham et al., 2015; Shwartz et al., 2016; Santus et al., 2014). Another line of work in representing the compositionality of meaning of words using syntactic structures(like Adjective-Noun pairs) is another approach towards semantic relation representations. (Baroni and Zamparelli, 2010; Guevara, 2010).

The kind of relation representations we aim at learning are meant to encode general relational knowledge and are produced in an unsupervised way, even though it can be useful for identification of specific relations like hypernymy and for relation extraction from text occurrences (Jameel et al., 2018). The latter paper documents a model that produces word pair embeddings by concatenating Glove-based word vectors with relation embeddings trained to predict the contexts in which the two words of the pair co-occur. The main issue with Jameel et al.'s models is scalability: as the authors admit, it is prohibitively expensive to collect all the data needed to train all the relation embeddings. Instead, their implementation requires, for each individual word pair, going back to the training corpus via an inverse index and collecting the data needed to estimate the embedding of the pair. This strategy might not be efficient for practical applications.

3 Proposed Model

We propose a simple solution to the scalability problem inherent in word relation embedding learning from joint cooccurrence data, which also allows the model to generalize to word pairs that never occur together in the corpus, or occur too rarely to accumulate significant relational cues information. The model is trained in two steps.

First, we apply the skip-gram with negative sampling algorithm to learn relation vectors for pairs of nouns n_1, n_2 with high individual and joint occurrence frequencies. In our experiments, all word pairs with pair frequency more than 100 and its individual word frequency more than 500 are considered as *frequent pair*s. To estimate the **SkipRel** vector of the pair, we adapted the learning objective of skip-gram with negative sampling, maximizing

$$log\sigma(v_c'^T.u_{n_1:n_2}) + \Sigma_{i=1}^{k} \mathbb{E}_{c_i^* \sim P_n(c)}[log\sigma(-v_{c_i^*}'^T.u_{n_1:n_2})] \quad (1)$$

where $u_{n_1:n_2}$ is the SkipRel embedding of a word pair, v_c' is the embedding of a context word occurring between n_1 and n_2, and k is the number of negative samples.

High-quality SkipRel embeddings can only obtained for noun pairs that co-occur frequently. To allow the model to generalize to noun pairs that do not co-occur in our corpus, we estimated an inter-

polation $\tilde{u}_{n_1:n_2}$ of the word pair embedding

$$\tilde{u}_{n_1:n_2} = relU(Av_{n_1} + Bv_{n_2}) \quad (2)$$

where v_{n_1}, v_{n_2} are pretrained word embeddings for the two nouns and the matrices A, B encode systematic correspondences between the embeddings of a word and the relations it participates in. Matrices A, B were estimated using stochastic gradient descent with the objective of minimizing the square error for the SkipRel vectors of frequent noun pairs n_1, n_2

$$\frac{1}{\mid P \mid}\Sigma_{n_1:n_2 \in P}(\tilde{u}_{n_1:n_2} - u_{n_1:n_2}) \quad (3)$$

We call $\tilde{u}_{n_1:n_2}$ the generalized SkipRel embedding (g-SkipRel) for the noun pair n_1, n_2. **RelWord**, the proposed relation embedding, is the concatenation of the **g-SkipRel** vector $\tilde{u}_{n_1:n_2}$ and the **Diff** vector $v_{n_1} - v_{n_2}$.

4 Experimental setup

We trained relation vectors on the ukWAC corpus (Baroni et al., 2009) containing 2 bln tokens of web-crawled English text. SkipRel is trained on noun pair instances separated by at most 10 context tokens with embedding size of 400 and mini-batch size of 32. Frequency filtering is performed to control the size of pair vocabulary ($|P|$). Frequent pairs are pre-selected using pair and word frequency thresholds. For pretrained word embeddings we used the best model from Baroni et al. (2014).

The experimental setup is built and maintained on GPU clusters provided by GRID5000 (Cappello et al., 2005). The code for model implementation and evaluation is publicly available at `https://github.com/Chingcham/SemRelationExtraction`

5 Evaluation

If our relation representations are rich enough in the information they encode, they will prove useful for any relation classification task regardless of the nature of the classes involved. We evaluate the model with a supervised softmax classifier on 2 labeled multiclass datasets, BLESS (Baroni and Lenci, 2011) and EVALuation1.0 (Santus et al., 2015), as well as the binary classification EACL antonym-synonym dataset (Nguyen et al., 2017). BLESS set consists of 26k triples of concept and relata spanned across 8 classes of semantic relation and EVALuation1.0 has 7.5k datasets spanned across 9 unique relation types. From EACL_2017 dataset, we used a list of 4062 noun pairs.

Model	BLESS	EVAL	EACL
Diff	81.15	57.83	71.25
g-SkipRel	59.07	48.06	70.31
RelWord	80.94	**59.05**	**73.88**
Random	12.5	11.11	50
Majority	24.71	25.67	50.4

Table 1: Semantic relation classification accuracy

Since we aim at recognizing whether the information relevant for relation identification is present in the representations in an easily accessible form, we choose to employ a simple, one-layer SoftMax classifier. The classifier was trained for 100 epochs, and the learning rate for the model is defined through crossvalidation. L2 regularization is employed to avoid over-fitting and the l2 factor is decided through empirical analysis. The classifier is trained with mini-batches of size 16 for BLESS & EVALuation1.0 and 8 for EACL_2017. SGD is utilized for optimizing model weights.

We prove the efficiency of RelWord vectors, we contrast them with the simpler representations of (g-)SkipRel and to Diff, the difference of the two word vectors in a pair, which is a commonly used simple method. We also include two simple baselines: random choice between the classes and the constant classifier that always predicts the majority class.

6 Results

All models outperform the baselines by a wide margin (Table 1). RelWord model compares favorably with the other options, outperforming them on EVAL and EACL datasets and being on par with the vector difference model for BLESS. This result signifies a success of our generalization strategy, because in each dataset only a minority of examples had pair representations directly trained from corpora; most WordRel vectors were interpolated from word emeddings.

Now let us restrict our attention to word pairs that frequently co-occur (Table 2). Note that the composition of classes, and by consequence the majority baseline, is different from Table 1, so the accuracy figures in the two tables are not di-

Model	BLESS	EVAL	EACL
Diff	77.13	44.61	66.07
SkipRel	73.37	48.40	**83.03**
RelWord	**83.27**	**54.47**	79.46
Random	12.5	11.11	50
Majority	33.22	26.37	63.63

Table 2: Semantic relation classification accuracy for frequent pairs

pair	gold	Diff	RelWord
bottle, can	antonym	hasproperty	hasa
race, time	hasproperty	hasa	antonym
balloon, hollow	hasproperty	antonym	hasa
clear, settle	isa	antonym	synonym
develop, grow	isa	antonym	synonym
exercise, move	entails	antonym	isa
fact, true	hasproperty	antonym	synonym
human, male	isa	synonym	hasproperty
respect, see	isa	antonym	synonym
slice, hit	isa	antonym	synonym

Table 3: Ten random examples in which RelWord and Diff make different errors. In the first one, the two models make predictions of comparable quality. In the second one, Diff makes a more intuitive error. In the remaining examples, RelWord's prediction is comparatively more adequate.

rectly comparable. For these frequent pairs we can rely on SkipRel relation vectors that have been estimated directly from corpora and have a higher quality; we also use SkipRel vectors instead of g-SkipRel as a component of RelWord. We note that for these pairs the performance of the Diff method dropped uniformly. This presumably happened in part because the classifier could no longer rely on the information on relative frequencies of the two words which is implicitly present in Diff representations; for example, it is possible that antonyms have more similar frequencies than synonyms in the EACL dataset. For BLESS and EVAL, the drop in the performance of Diff could have happened in part because the classes that include more frequent pairs such as isA, antonyms and synonyms are inherently harder to distinguish than classes that tend to contain rare pairs. In contrast, the comparative effectiveness of RelWord is more pronounced after frequency filtering. The usefulness of relation embeddings is especially impressive for the EACL dataset. In this case, vanilla SkipRel emerges as the best model, confirming that word embeddings *per se* are not particularly useful for detecting the synonymy-antonymy distinction for this subset of EACL, getting an accuracy just above the majority baseline, while pair embeddings go a long way.

Finally, quantitative evaluation in terms of classification accuracy or other measures does not fully characterize the relative performance of the models; among other things, certain types of misclassification might be worse than others. For example, a human annotator would rarely confuse synonyms with antonyms, while mistaking `has_a` for `has_property` could be a common point of disagreement between annotators. To do a qualitative analysis of errors made by different models, we selected the elements of EVAL test partition where Diff and RelWord make distinct predictions that are both different from the gold standard label. We manually annotated for each of the 53 examples of this kind, which model is more a acceptable according to a human's judgment. In a majority of cases (28) the WordRel model makes a prediction that is more human-like than that of Diff. For example, WordRel predicts that *shade* is part of *shadow* rather than its synonym (gold label); indeed, any part of a shadow can be called *shade*. The Diff model in this case and in many other examples bets on the antonym class, which does not make any sense semantically; the reason why `antonym` is a common false label is probably that it is simply the second biggest class in the dataset. The examples where Diff makes a more meaningful error than RelWord are less numerous (6 out of 53). There are also 15 examples where both system's predictions are equally bad (for example, for *Nice,France* Diff predict `isa` label and WordRel predicts `synonym`) and 4 examples where the two predictions are equally reasonable. For more examples, see Table 3. We note that sometimes our model's prediction seems more correct than the gold standard, for example in assigning `hasproperty` rather than `isa` label to the pair *human, male*.

7 Conclusion

The proposed model is simple in design and training, learning word relation vectors based on co-occurrence with unigram contexts and extending to rare or unseen words via a non-linear mapping. Despite its simplicity, the model is capable of capturing lexical relation patterns in vector representations. Most importantly, RelWord extends straightforwardly to novel word pairs in a

manner that does not require recomputing cooccurrence counts from the corpus as in related approaches (Jameel et al., 2018). This allows for an easy integration of the pretrained model into various downstream applications.

In our evaluation, we observed that learning word pair relation embeddings improves on the semantic information already present in word embeddings. With respect to certain semantic relations like synonyms, the performance of relation embedding is comparable to that of word embeddings but with an additional cost of training a representation for a significant number of pair of words. For other relation types like antonyms or hypernyms, in which words differ semantically but share similar contexts, learned word pair relation embeddings have an edge over those derived from word embeddings via simple subtraction. While in practice one has to make a choice based on the task requirements, it is generally beneficial to combine both types of relation embeddings for best results in a model like RelWord.

Our current model employs pretrained word embeddings and learns the word pair embeddings and a word-to-relation embedding mapping separately. In the future, we plan to train a version of the model end-to-end, with word embeddings and the mapping trained simultaneously. As literature suggests (Hashimoto et al., 2015; Takase et al., 2016), such joint training might not only benefit the model but also improve the performance of the resulting word embeddings on other tasks.

Acknowledgments

This research is supported by CNRS PEPS grant ReSeRVe. We thank Roberto Zamparelli, Germán Kruszewski, Luca Ducceschi and anonymous reviewers who gave feedback on previous versions of this work.

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

Tuan Luu Anh, Yi Tay, Siu Cheung Hui, and See Kiong Ng. 2016. Learning term embeddings for taxonomic relation identification using dynamic weighting neural network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 403–413.

Marco Baroni and Alessandro Lenci. 2011. How we blessed distributional semantic evaluation. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, GEMS '11, pages 1–10, Stroudsburg, PA, USA. Association for Computational Linguistics.

Marco Baroni and Roberto Zamparelli. 2010. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 1183–1193, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

Marco Baroni, Georgiana Dinu, and Germn Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference*, volume 1, pages 238–247, 06.

Samuel Ryan Bowman. 2016. *Modeling natural language semantics in learned representations*. Ph.D. thesis, Ph. D. thesis, Stanford University.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 49:1–47.

Franck Cappello, Eddy Caron, Michel J. Dayd, Frdric Desprez, Yvon Jgou, Pascale Vicat-Blanc Primet, Emmanuel Jeannot, Stphane Lanteri, Julien Leduc, Nouredine Melab, Guillaume Mornet, Raymond Namyst, Benjamin Qutier, and Olivier Richard. 2005. Grid'5000: a large scale and highly reconfigurable grid experimental testbed. In *GRID*, pages 99–106. IEEE Computer Society.

Miao Fan, Kai Cao, Yifan He, and Ralph Grishman. 2015. Jointly embedding relations and mentions for knowledge population. *arXiv preprint arXiv:1504.01683*.

Ruiji Fu, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Learning semantic hierarchies via word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1199–1209.

Emiliano Guevara. 2010. A regression model of adjective-noun compositionality in distributional semantics. In *Proceedings of the 2010 Workshop on GEometrical Models of Natural Language Semantics*, GEMS '10, pages 33–37, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zellig Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Kazuma Hashimoto, Pontus Stenetorp, Makoto Miwa, and Yoshimasa Tsuruoka. 2015. Task-oriented learning of word embeddings for semantic relation classification. *arXiv preprint arXiv:1503.00095*.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. Technical Report S2K-92-09.

Aurélie Herbelot and Eva Maria Vecchi. 2015. Building a shared world: Mapping distributional to model-theoretic semantic spaces. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 22–32.

Shoaib Jameel, Zied Bouraoui, and Steven Schockaert. 2018. Unsupervised learning of distributional relation vectors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23–33. Association for Computational Linguistics.

German Kruszewski, Denis Paperno, and Marco Baroni. 2015. Deriving boolean structures from distributional vectors. *Transactions of the Association for Computational Linguistics*, 3:375–388.

Omer Levy and Yoav Goldberg. 2014. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the eighteenth conference on computational natural language learning*, pages 171–180.

Tal Linzen. 2016. Issues in evaluating semantic spaces using word analogies. *arXiv preprint arXiv:1606.07736*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. 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: Volume 2 - Volume 2*, ACL '09, pages 1003–1011, Stroudsburg, PA, USA. Association for Computational Linguistics.

Thien Huu Nguyen and Ralph Grishman. 2015. Relation extraction: Perspective from convolutional neural networks. In Phil Blunsom, Shay B. Cohen, Paramveer S. Dhillon, and Percy Liang, editors, *VS@HLT-NAACL*, pages 39–48. The Association for Computational Linguistics.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017. Distinguishing Antonyms and Synonyms in a Pattern-based Neural Network. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 76–85, Valencia, Spain.

Masataka Ono, Makoto Miwa, and Yutaka Sasaki. 2015. Word embedding-based antonym detection using thesauri and distributional information. In *HLT-NAACL*, pages 984–989.

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.

Nghia The Pham, Angeliki Lazaridou, Marco Baroni, et al. 2015. A multitask objective to inject lexical contrast into distributional semantics. 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)*, volume 2, pages 21–26.

Stephen Roller and Katrin Erk. 2016. Relations such as hypernymy: Identifying and exploiting hearst patterns in distributional vectors for lexical entailment. *CoRR*, abs/1605.05433.

Ina Rösiger, Arndt Riester, and Jonas Kuhn. 2018. Bridging resolution: Task definition, corpus resources and rule-based experiments. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3516–3528.

Enrico Santus, Qin Lu, Alessandro Lenci, and Churen Huang. 2014. Unsupervised antonym-synonym discrimination in vector space.

Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. Evalution 1.0: an evolving semantic dataset for training and evaluation of distributional semantic models. In *Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications*, pages 64–69.

Vered Shwartz, Enrico Santus, and Dominik Schlechtweg. 2016. Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection. *arXiv preprint arXiv:1612.04460*.

Sho Takase, Naoaki Okazaki, and Kentaro Inui. 2016. Modeling semantic compositionality of relational patterns. *Engineering Applications of Artificial Intelligence*, 50:256–264.

Kun Xu, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2015. Semantic relation classification via convolutional neural networks with simple negative sampling. *CoRR*, abs/1506.07650.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of Machine Learning Research*, 3:1083–1106.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, Jun Zhao, et al. 2014. Relation classification via convolutional deep neural network. In *COLING*, pages 2335–2344.

A NLP-based Analysis of Reflective Writings by Italian Teachers

Giulia Chiriatti•, Valentina Della Gala⋆, Felice Dell'Orletta◇, Simonetta Montemagni◇, Maria Chiara Pettenati⋆, Maria Teresa Sagri⋆, Giulia Venturi◇

•Università di Pisa
giuliachiriatti@gmail.com

⋆Istituto Nazionale Documentazione, Innovazione, Ricerca Educativa (INDIRE)
{v.dellagala,mc.pettenati,t.sagri}@indire.it

◇Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC-CNR) - ItaliaNLP Lab
{nome.cognome}@ilc.cnr.it

Abstract

English. This paper reports first results of a wider study devoted to exploit the potentialities of a NLP-based approach to the analysis of a corpus of reflective writings on teaching activities. We investigate how a wide set of linguistic features allows reconstructing the linguistic profile of the texts written by the Italian teachers and predicting whether are reflective.

Italiano. *L'articolo descrive i primi risultati di uno studio più ampio che impiega strumenti e metodi di analisi e classificazione automatica del testo per descrivere le caratteristiche linguistiche di un corpus di documenti scritti dai neoassunti nella scuola italiana che riflettono su una specifica esperienza didattica.*

1 Introduction

Since 2014, the "National Institute for Documentation, Innovation and Educational Research" (INDIRE) manages for the Ministry of Education (MIUR) the *induction program* of the Italian Newly Qualified Teachers (NQTs), i.e. the induction phase of teachers professional development that aims to support teachers in their transition from their initial teacher education into working life in schools. Experimented for the first time in 2014, it became effective starting in 2015 with the DM 850/2015.[1] The program involves all new hiring teachers from primary to secondary school for a total of 130,000 NQTs committed in the last 3 years. The underlying theoretical framework developed by INDIRE, MIUR and University of Macerata is based on the alternation of laboratorial and traditional classroom activities with documentation and reflection activities. The purpose is "to influence practices through a process that alternates between moments of immersion and distancing, which are actualised in *When I teach* and *When I reconsider my teaching to think of what happened*" (Magnoler et al., 2016). An on-line environment developed and managed by INDIRE[2] was set up to support teachers to reflect about and document their educational and professional activities (see Figure 1) during the induction program. All evidences of the instructional tasks (surveys, writing tasks, lesson plans, instructional materials, etc.) are collected in the e-portfolio and printed by the teachers for the final exam. An yearly monitoring of teachers activities is carried on by INDIRE to assess the effectiveness of the whole induction program, as well as of the single instructional tasks. It is aimed to modify, whenever needed, the program in order to improve stakeholders' scaffolding to the newly qualified teachers and lastly teachers' professional development.

Figure 1: The on-line environment collecting the e-portfolio of the newly qualified teachers.

In this paper, we report first results of an ongoing study devoted to investigate the potentialities offered by Natural Language Processing methods and tools for the analysis of the NQTs e-

[1] http://neoassunti.indire.it/2018/files/DM_850_27_10_2015.pdf

[2] The e-portfolio is available at http://neoassunti.indire.it/2018/

portfolio. We consider in particular the documents written by the 26,526 teachers hired in the 2016/17 school year. Many protocols (or models) have been proposed to assess reflection in teachers writing, e.g. (Sparks-Langer et al., 1990; Hatton and Smith, 1995; Kember et al., 2008; Larrivee, 2008; Harland and Wondra, 2011). These models rely on features that suggest either different levels of reflection (means focused on the depth of reflection) or content of reflection (focused on the breadth of reflection), and usually they have found to mix features of both classes (depth and breadth) (Ullmann, 2015). We rather focus here on the analysis of the *form* to study which are the main linguistic phenomena, distinguishing reflective from non reflective writings. Specifically, we devised a methodology devoted to investigate whether and to which extent a wide set of linguistic features automatically extracted from texts can be exploited to characterize NQTs' reflective writings.

Our contribution: *i)* we collect a corpus of reflective writings manually annotated by experts in the learning science domain and classified with respect to different types of reflectivity; *ii)* we detect a wide set of linguistically phenomena, characterizing the collected writings; *iii)* we report the first results of an automatic classification experiment to assess which features contribute more in the automatic prediction of reflexivity.

2 Defining reflection

Within the teaching and teacher education domain, a very large amount of studies have been dedicated to conceptualization and analysis of teachers reflection and teachers' reflective practice. Dewey (1933), Van Manen (1977), Schon (1984; Schon (1987; Schon (1991), Mezirow (1990) are among the main references. The attention on reflective thinking in the teachers education field has increased starting from the 80s as a reaction to the overlay technical view of teaching. Scholars have intensely studied reflection as a concept, detected more levels and types of reflection, how it works during and after professional teachers' practice, its role and purpose in teachers' professional development, and how it can be embedded in the curriculum of teachers preparation or professional development, and which techniques may be used to promote it (groups of discussion, readings, oral interview, action research projects, writing tasks, etc). In his seminal work "How we think", Dewey provides the most shared definition of reflective thinking as applied in the educational field: reflection may be seen as an "active, persistent, and careful consideration of any belief or supposed form of knowledge in the light of the grounds that support it and the further conclusions to which tends". Hence, reflection is a systematic process of thinking that happens only if related to actual experiences, and includes observation of conditions and references to different pieces of knowledge, (i.e. references to previous experiences, domain knowledge, common sense knowledge, etc.), in order to respond to a dilemma (Mezirow, 1990). Teachers' educators have extensively employed writing tasks, such as writing structured or unstructured journals, portfolios, essays, blogs, open-ended questions to foster reflection both in pre-service and experienced teachers. Operational definitions of reflectivity proposed to develop schemes for assessing it are focused on identifying the presence of "reflective content" in teachers' writing, or how deep the reflection is.

Based on these premises, we are currently developing a reflection assessment schema suitable to describe properly the peculiarities of the Italian teachers' reflective writings written in the framework of the 2016/17 induction program. The schema designed so far, reported in Table 1, was devised according to the following criteria: a writing is reflective if it i) makes direct references to experienced teaching activity, ii) involves several topics (content/pedagogical knowledge) and references to previous experiences, classroom management, learners needs, iii) includes premises analysis (theoretical, context-related, personal) iv) debates a problem (a dilemma), a doubt, v) has an output: it sums up what was learned, sketches future plans, gives a new insight and understanding for immediate or future actions.

3 The Corpus

The corpus of NQTs reflective writings is part of the wider collection of documents written by the 26,526 teachers engaged in the 2016/17 INDIRE induction program. The whole corpus includes all texts written in two of the seven activities of the e-portfolio: *Didactic Activity 1* and *2* (DA) for a total of 265,200 texts. During these two activities, teachers were supported by guiding questions designed by INDIRE experts to help them to understand the consistency of the planned and acted

Type of reflectivity	Description	Example
No reflection	Simple writing that merely describes what happened during the teaching activity, no doubts or clues of an inquiry attitude are shown	I contenuti presentati sono stati acquisiti e gli alunni intervistati si sono dimostrati soddisfatti dell'intervento e del parere personale che hanno potuto esprimere sull'argomento di discussione.
General considerations and understanding	Writing shows weak links to the actual teaching experience, it is conducted at a distance from the phenomena of interest. It can include general thoughts and considerations	Per rispondere alla domanda circa la possibilità di migliorare l'attività affrontata, dirò innanzitutto che ritengo sempre possibile migliorare le proprie prestazioni. Sono convinta che l'esperienza sia una grande alleata e che, col tempo, si cresca, ci si arricchisca e si migliori.
Descriptive reflection	Writing includes considerations on actual classroom actions/events and some kind of knowledge base but doesn't clearly refer to any "problems", doubt or dilemma	Credo che la scelta più efficace sia stata quella della valutazione tra pari. In particolare, durante la fase della premiazione del concorso di poesia, un alunno per classe si è recato nell'altra scuola e ha tenuto un discorso introduttivo alla premiazione, nonché gestito la stessa in autonomia. Questo, a mio avviso, ha fatto sentire gli studenti i veri protagonisti del loro lavoro e ha favorito la motivazione, intrinseca ed estrinseca. Le consegne sono sempre state fornite in modo chiaro, ma hanno necessitato diverse ripetizioni per essere assimilate.
Reflection	Writing discusses problems, doubts and refers to some kind of action. It may report a reflective practice. There could be evidences of a change on teachers' attitude or acquiring new insights due to the problems faced	In realtà, mi sono accorta che solo pochi di loro erano capaci di dare una spiegazione adeguata (anche dal punto di vista formale) e soprattutto non riuscivano a trovare esempi calzanti se non con l'aiuto del libro di testo. Questo momento di ricognizione ha portato via quasi il doppio del tempo che avevo previsto, ma è comunque stato molto utile per accelerare il loro compito di ricerca durante l'analisi del nuovo testo proposto. Li ho stimolati a chiarire ogni dubbio e grazie anche alle loro domande credo che gli argomenti siano stati davvero appresi da tutti gli studenti, anche da chi di solito ha più difficoltà o da chi normalmente partecipa meno. È stata una lezione che li ha molto coinvolti nonostante si trattasse di una lezione piuttosto "tradizionale", perché mi hanno detto che questo sarebbe servito loro anche per lo studio di altre materie e soprattutto in vista dell'esame.

Table 1: Annotation schema of reflectivity.

teaching activities. For DA 1 and 2 they wrote 5 short texts as answers to 5 different groups of questions. The first 4 groups provide guidance for teachers to write general reflections only on the *design* of their teaching activity; the fifth group is meant to guide NQTs towards an overall reflection on their whole teaching experience, i.e. both the design and the real teaching activity, also including classroom assessment techniques.

We focused here on the answers to this latter group of questions that were devised in order to encourage teachers to reflect on the following issues: *i)* differences and similarities between the designed and achieved activities, *ii)* the most effective choices adopted, also including classroom assessment techniques, *iii)* how the activity could be improved, *iv)* the role played by the tutor and documentation practices. We considered in particular a subset of this group of answers that were annotated by 3 experts in the learning science domain according to the reflectivity annotation scheme described in Section 2 (see Table 2). The agreement between the three annotators was calculated using the Fleiss' kappa test and we obtained a k=0.66, i.e. substantial agreement.

Reflectivity	n. answers	n. sent.	n. tokens
No reflection	185	348	9,784
Rhetoric	35	91	3,140
Reflection	217	609	21,686
Radical reflection	36	149	5,326
TOTAL	**473**	**1,197**	**39,936**

Table 2: Corpus of NQTs reflective writings annotated for different types of reflectivity.

4 Linguistic Features and Reflectivity

The annotated corpus was tagged by the part-of-speech tagger described in Dell'Orletta (2009) and dependency-parsed by the DeSR parser (Attardi

et al., 2009). This allowed to extract a wide set of multilevel features, i.e. raw text, lexical, morpho-syntactic and syntactic, fully described by Dell'Orletta et al. (2013). They was used to reconstruct the linguistic profile of reflective writings and to carry out a first classification experiment aimed at predicting whether a text is reflective.

4.1 Distribution of Linguistic Features

Table 3 shows a selection of the features that vary significantly *i)* between reflective and non-reflective answers (column *Reflectivity*) and *ii)* among the different types of reflectivity we considered (column *Types of Reflectivity*)[3]. The analysis of variance was computed in the first case using the Wilcoxon Rank-sum test for paired samples, while in the second case we used the Kruskal-Wallis test since we aimed to assess the different distribution of features in the 4 classes.

In both cases, features from all levels of analysis resulted to be significant. If we consider the first ten most discriminative features, reflective writings resulted to be longer in terms of number of words and sentences, they are characterized by longer sentences and by a lower Type/Token Ratio; they contain an higher number of verbal heads and of embedded complement 'chains' (governed by a nominal head). Interestingly, they mostly contain linguistic phenomena typically related to syntactic complexity, for example they are characterized by *i)* an higher use of verbal modification (e.g. higher % of adverbs, of auxiliary and modal verbs), *ii)* more complex verbal predicate structures (e.g. higher average verbal arity, calculated as the number of instantiated dependency links sharing the same verbal head), *iii)* more extensive use of subordination (e.g. higher % of subordinate clauses also embedded in deep chains), *iv)* features related to a non canonical word order (e.g. higher % of pre-verbal objects and post-verbal subjects), *v)* longer dependency links and higher parse trees, two features related to sentence length. On the contrary, non reflective NQTs' answers contain an higher level of lexical complexity: they have an higher Type/Token Ratio, a lower percentage of "Fundamental words", i.e. very frequent words according to the classification proposed by De Mauro (2000) in the *Basic Italian Vocabulary* (BIV), and an higher percentage of "High usage words".

[3]The full list of ranked features is contained in Appendix.

If we focus on the linguistic profile of the different types of reflective writings, we can observe that answers annotated as *Reflection* and *Radical reflection* are mostly characterized by features typically related to structural complexity. This is particular the case of *Radical reflection* answers that are longer in terms of number of sentences and words; they have more complex verbal predicates (e.g. an higher % of adverbs and of an implicit mood such as gerundive that can be more ambiguous with respect to the referential subject), more complex use of subordination (e.g. average length of 'chains' of embedded subordinate clauses), long distance constructions (length of dependency links), non canonical constructions (post-verbal subject). The higher % of demonstrative pronouns and determiners can be related to one of the most representative characteristic of reflection, i.e. the direct reference to real life. On the contrary, they contain a simpler use of lexicon, e.g. a lower Type/Token ratio and an higher percentage of "Fundamental words".

4.2 Prediction of Reflectivity

Table 4 reports the results of the automatic classification experiment we devised in order to predict whether a text is reflective. We built a classifier based on LIBLINEAR (Fan et al., 2008) as machine learning library trained using the LIBLINEAR L2-regularized L2-loss support vector classification function. We followed a 5-fold cross-validation process and relied on a training set of 370 answers balanced between the reflective and non reflective texts, since the under sampling technique has been proofed to improve classification performance on unbalanced datasets (Qazi and Raza, 2012). The performance was calculated in terms of F-score in the correct classification of *non reflective* (*0* in the table) or of *reflective* (*1*) writings. We used different classification models: the *Raw text* one uses only raw text features, the *Lexical* one uses the distribution of the lexicon belonging to the *Basic Italian Vocabulary* and up to bi-grams of words, the *Morpho-syntactic* one uses the unigram of part-of-speech and verbal morphology features, the *All features* model uses all the considered features including the syntactic ones. A very competitive baseline was computed: it exploits the distribution of unigrams of words (*Unigrams*). As it can be seen, the model that uses all the considered features resulted to be the best

Feature	Ranking position		Avg. Feature Value in different types of (non)reflective texts			
	Reflectivity	Types of Reflectivity	No reflection	Rhetoric	Reflection	Radical reflection
Raw text features:						
Avg sentence length	**10**	**11**	27.97	35.9	**38.6**	38.2
Avg number of sentences	**9**	**7**	1.88	2.6	2.81	**4.14**
Avg number of words	**1**	**1**	52.89	89.71	99.94	**147.94**
Lexical features:						
Type/token ratio (100 token)	**8**	**9**	**0.78**	0.71	0.7	0.69
% of "Fundamental words" of *BIV*	*62*	*86*	74.15	75.57	77.01	**77.92**
% of "High usage words" of *BIV*	*92*	**38**	**19.35**	15.79	15.71	14.92
% of "High availability words" of *BIV*	*58*	*68*	9.72	**12.8**	10.78	10.69
Morpho–syntactic features:						
% of adjectives	*71*	*87*	7.29	**9.16**	7.72	7.93
% of possessive adjectives	*67*	**43**	1.08	**2**	0.97	0.93
% of adverbs	**42**	**46**	3.95	3.93	4.82	**5.29**
% of prepositions	*51*	*82*	15.11	**17.08**	16.61	16.05
% of demonstrative pronouns	**36**	**34**	0.43	0.65	0.58	**0.78**
% of demonstrative determiners	**35**	**30**	0.35	0.66	0.42	**0.6**
% of determinative articles	**30**	**41**	**8.29**	6.89	6.81	7.07
% of subordinative conjunctions	*69*	*63*	0.94	0.68	0.98	**1.27**
% of sentence boundary punctuation	**12**	**12**	**4.17**	2.99	2.86	2.92
% of auxiliary verbs	**25**	**27**	**6.66**	4.01	4.92	4.48
% of modal verbs	**40**	**40**	0.69	**1.06**	0.78	0.97
% of verbs – subjective mood	*72*	**39**	1.16	1.29	**2.55**	1.53
% of verbs – infinitive mood	**28**	**36**	19.11	**27.48**	25.03	25.75
% of verbs – gerundive mood	**37**	**45**	5.54	6.06	6.51	**6.73**
% of verbs – indicative mood	**38**	*58*	10.46	**14.76**	11.74	12.91
% of verbs – third person singular	**20**	**15**	8.2	18.76	14.92	**19.3**
% of verbs – third person plural	*80*	*91*	6.14	**10.83**	8.04	7.67
% of verbs – imperfect tense	*78*	**35**	7.18	1.55	9.72	**13.75**
Syntactic features:						
% of dependency types – auxiliary	**24**	**25**	**6.65**	3.98	4.88	4.41
% of dependency types – object	*44*	*59*	4.22	4.7	5.06	**5.6**
% of dependency types – preposition	*55*	*81*	15.15	**17.33**	16.6	16.09
% of dependency types – subordinate clause	*60*	*62*	0.99	0.78	1.03	**1.22**
% of dependency types – subject	*46*	*83*	**4.62**	3.62	3.77	3.74
Avg number of verbal heads	**2**	**3**	52.89	89.71	99.94	**147.94**
Avg number of embedded complement chains	**4**	**4**	9.72	**12.8**	10.78	10.69
Length of 'chains' of embedded subordinate clauses (avg)	**19**	**21**	0.48	0.69	0.86	**0.95**
Maximum length of dependency links (avg)	**16**	**19**	10.26	12.71	14.16	**14.8**
Parse tree depth (avg)	**21**	**24**	7.86	9.73	9.56	**9.65**
Arity of verbal predicates (avg)	**13**	**13**	3.62	4.46	**4.89**	4.74
% of pre-verbal objects	*52*	**42**	4.84	**9.71**	7.59	4.81
% of post-verbal subject	*86*	*84*	10.65	11.17	10.64	**17.07**
% of subordinate clauses in post-verbal position	**23**	**16**	52.21	76.57	78.97	**97.71**

Table 3: Feature ranking position characterizing *i)* reflective vs. non reflective texts and *ii)* different types of reflective texts and average value of feature distribution in the different types of reflective texts. Ranking positions with p <0.001 are marked in italics and with p <0.05 in boldface.

one. On the contrary, the model relying on very simple types of features (raw text features) that capture how much teachers have written achieves the worst results. We also carried out a very preliminary experiment to classify the three different types of reflective writings but it produced unsatisfactory results due to the unbalanced distribution of answers in the reflective classes. As expected, a balanced experiment yielded very low accuracies since we used very few data.

Features	F1 0	F1 1	Tot F1
Raw text	58.4	69.86	64.13
Lexical	78.58	77.53	78.05
Morpho-syntactic	74.87	75.18	75.02
All features	**79.31**	**79.01**	**79.16**
Baseline (unigrams)	75.16	74.84	75.00

Table 4: Classification of reflective vs. non reflective writings using different models of features.

5 Conclusions and current developments

We reported first results of a on-going study devoted to reconstruct the linguistic profile of a corpus of reflective writings by Italian newly recruited teachers that we collected for the specific purpose of this paper. We are currently enlarging the corpus with new manually annotated data to improve the accuracy of the automatic classification of different types of reflectivity.

References

G. Attardi, F. Dell'Orletta, M. Simi and J. Turian. 2009. Accurate dependency parsing with a stacked

multilayer perceptron. *Proceedings of Evalita'09, Evaluation of NLP and Speech Tools for Italian* , Reggio Emilia, December.

D. Boud and D. Walker. 2013. *Reflection: Turning Experience into Learning*. RoutledgeFalmer.

C.C. Chang and C.J. Lin. 2001. LIBSVM: a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm

F. Dell'Orletta. 2009. Ensemble system for Part-of-Speech tagging. *Proceedings of Evalita'09, Evaluation of NLP and Speech Tools for Italian* , Reggio Emilia, December.

F. Dell'Orletta, S. Montemagni and G. Venturi. 2013. Linguistic profiling of texts across textual genre and readability level. An exploratory study on Italian fictional prose. *Proceedings of the Recent Advances in Natural Language Processing Conference (RANLP-2013)*.

T. De Mauro. 2000. *Grande dizionario italiano dell'uso (GRADIT)*. Torino, UTET.

J. Dewey. 1933. *How we think: a restatement of the relation of reflective thinking to the educative process*. D.C. Heath and company.

R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X. Wang, and C.-J. Lin. 2008. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9:1871–1874.

DJ. Harland and JD. Wondra 2011. Presercice Teachers' Reflection on Clinical Experiences: A Comparison of Blog and Final Paper Assignments. *Journal of Digital Learning in Teacher Education*, Vol. 27(4).

N. Hatton and D. Smith. 1995. Reflection in teacher education: Towards definition and implementation. *Teaching and Teacher Education*, Vol. 11(1).

D. Kember, J. McKey, K. Sinclair, FKY Wong 2008. A forur category scheme for coding and assessing the level of reflection in written work. *Assessment and Evaluation in Higher Education*, Vol. 25(4).

B. Larrivee 2008. Development of a tool to assess teachers' level of reflective practice. *Reflective Practice*, Vol. 9(3).

P. Magnoler, GR. Mangione, MC. Pettenati, A. Rosa, PG. Rossi. 2016. Induction models and teachers professional development. *Journal of e-Learning and Knowledge Society*, Vol. 12(3).

J. Mezirow. 1990. *Fostering critical reflection in adulthood: a guide to transformative and emancipatory learning*. Jossey-Bass Publishers.

N. Qazi and K. Raza. 2012. Effect of Feature Selection, SMOTE and under Sampling on Class Imbalance Classification. *Proceedings of the 2012 UKSim 14th International Conference on Modelling and Simulation*, pp. 145-150.

D.A. Schon. 1984. *The Reflective Practitioner: How Professionals Think In Action*. Basic Books.

D.A. Schon. 1987. *Educating the Reflective Practitioner*. Jossey-Bass.

D.A. Schon. 1991. *The reflective turn: Case studies in and on educational practice*. Teachers College Press.

GM. Sparks-Langer, GM. Simmons, M. Pasch, A. Colton, A. Starko. 1990. Reflective pedagogical thinking: How can we promote it and measure it? *Journal of Teacher Education*, Vol. 41(5).

T. D. Ullmann. 2015. *Automated detection of reflection in texts. A machine learning based approach*. The Open University.

T. D. Ullmann. 2015. *Keywords of written reflection - a comparison between reflective and descriptive datasets*. Proceedings of the 5th Workshop on Awareness and Reflection in Technology Enhanced Learning.

M. Van Manen. 1977. Linking Ways of Knowing with Ways of Being Practical. *Curriculum Inquiry*, Vol. 6(3).

H.C. Waxman et al. 1987. *Images of Reflection in Teacher Education*. Summaries of papers presented at a National Conference on Reflective Inquiry in Teacher Education, Houston.

Feature	Ranking position		Avg. Feature Value in different types of (non)reflective texts			
	Reflectivity	Types of Reflectivity	No reflection	Rhetoric	Reflection	Radical reflection
Raw text features:						
Avg sentence length	**10**	**11**	27.97	35.9	**38.6**	38.2
Avg number of sentences	**9**	**7**	1.88	2.6	2.81	**4.14**
Avg number of tokens	**1**	**1**	52.89	89.71	99.94	**147.94**
Lexical features:						
Type/token ratio (first 100 lemma)	**8**	**9**	**0.78**	0.71	0.7	0.69
Type/token ratio (first 200 lemma)	*6*	*6*	**0.77**	0.68	0.67	0.64
% of "Fundamental words" of *BIV*	*62*	*86*	74.15	75.57	77.01	**77.92**
% of "High usage words" of *BIV*	*92*	**38**	**19.35**	15.79	15.71	14.92
% of "High availability words" of *BIV*	*58*	*68*	9.72	**12.8**	10.78	10.69
Morpho–syntactic features:						
Lexical density	**64**	**96**	0.54	0.55	0.55	**0.56**
% of adjectives	*71*	*87*	7.29	**9.16**	7.72	7.93
% of possessive adjectives	*67*	**43**	1.08	**2**	0.97	0.93
% of adverbs	**42**	**46**	3.95	3.93	4.82	**5.29**
% of negative adverbs	**54**	**53**	0.64	0.38	0.64	**0.65**
% of determiners	**63**	**88**	1.19	1.19	1.28	**1.43**
% of demonstrative determiners	**35**	**30**	0.35	0.66	0.42	**0.6**
% of indefinite determiners	**74**	**71**	0.8	0.47	**0.83**	0.8
% of prepositions	*51*	*82*	15.11	**17.08**	16.61	16.05
% of articles	*93*	none	**9.36**	8.34	8.38	**8.64**
% of demonstrative pronouns	**36**	**34**	0.43	0.65	0.58	**0.78**
% of personal pronouns	**89**	**99**	0.29	**0.39**	0.32	0.24
% of relative pronouns	*39*	**56**	1.17	1.16	1.48	**1.55**
% of determinative articles	**30**	**41**	**8.29**	6.89	6.81	7.07
% of subordinative conjunctions	*69*	*63*	0.94	0.68	0.98	**1.27**
% of single commas or hyphens	*27*	*33*	3.55	4.7	4.67	**5.26**
% of numbers	**87**	**67**	0.22	0.19	**0.4**	0.29
% of sentence boundary punctuation	**12**	**12**	**4.17**	2.99	2.86	2.92
% of verbs	**48**	**70**	**20.51**	17.71	18.52	17.91
% of auxiliary verbs	**25**	**27**	**6.66**	4.01	**4.92**	4.48
% of modal verbs	**40**	**40**	0.69	**1.06**	0.78	0.97
% of verbs – subjective mood	*72*	**39**	1.16	1.29	**2.55**	1.53
% of verbs – infinitive mood	**28**	**36**	19.11	**27.48**	25.03	25.75
% of verbs – gerundive mood	**37**	**45**	5.54	6.06	6.51	**6.73**
% of verbs – indicative mood	**38**	*58*	10.46	**14.76**	11.74	12.91
% of verbs – third person singular	**20**	**15**	8.2	18.76	14.92	**19.3**
% of verbs – third person plural	*80*	*91*	6.14	**10.83**	8.04	7.67
% of verbs – imperfect tense	*78*	**35**	7.18	1.55	9.72	**13.75**
Syntactic features:						
% of syntactic roots	*14*	*14*	**4.57**	3.06	3.36	3.21
% of dep–auxiliary	**24**	**25**	**6.65**	3.98	4.88	4.41
% of dep–nominal/clausal argument	**61**	**98**	2.36	**3.08**	2.8	2.41
% of dep–indirect complement	**66**	**61**	0.46	0.62	0.5	0.48
% of dep–locative complement	**47**	*31*	0.07	0.21	**0.34**	0.14
% of dep–temporal complement	*41*	*28*	0.16	0.3	0.28	**0.41**
% of dep–nominal/clausal modifier	**45**	**73**	15.88	17.25	17.07	**17.7**
% of dep–relative modifier	*32*	*32*	1.18	1.1	1.46	**1.8**
% of dep–object	*44*	*59*	4.22	4.7	5.06	**5.6**
% of dep–preposition	*55*	*81*	15.15	**17.33**	16.6	16.09
% of dep–subordinate clause	*60*	*62*	0.99	0.78	1.03	**1.22**
% of dep–subject	*46*	*83*	**4.62**	3.62	3.77	3.74
Avg number of verbal heads	**2**	**3**	52.89	89.71	99.94	**147.94**
Avg number of embedded complement chains	**4**	**4**	9.72	**12.8**	10.78	10.69
Length of 'chains' of embedded subordinate clauses (avg)	**19**	**21**	0.48	0.69	0.86	**0.95**
Length of dependency links (avg)	*15*	*18*	2.09	2.3	2.4	**2.42**
Maximum length of dependency links (avg)	**16**	**19**	10.26	12.71	14.16	**14.8**
Parse tree depth (avg)	**21**	**24**	7.86	9.73	9.56	**9.65**
Arity of verbal predicates (avg)	**13**	**13**	3.62	4.46	**4.89**	4.74
% of verbal roots	**57**	*29*	**0.96**	**0.95**	0.9	0.84
% of verbal roots with explicit subj	**70**	**65**	67.92	**73.76**	59.05	60.79
% of finite complement clauses	**83**	**95**	19.85	17.19	23.08	**27.64**
% of infinite complement clauses						
% of pre-verbal objects	*52*	**42**	4.84	**9.71**	7.59	4.81
% of post-verbal subject	*86*	*84*	10.65	11.17	10.64	**17.07**
% of subordinate clauses in post-verbal position	**23**	**16**	52.21	76.57	78.97	**97.71**

Table 5: **Appendix A**: Full list of feature ranking positions characterizing *i)* reflective vs. non reflective texts and *ii)* different types of reflective texts and average value of feature distribution in the different types of reflective texts. Ranking positions with p <0.001 are marked in italics and with p <0.05 in boldface. Features which were not selected during ranking have no rank.

Sentences and Documents in Native Language Identification

Andrea Cimino, Felice Dell'Orletta, Dominique Brunato, Giulia Venturi
Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC–CNR)
ItaliaNLP Lab - *www.italianlp.it*
{name.surname}@ilc.cnr.it

Abstract

English. Starting from a wide set of linguistic features, we present the first in depth feature analysis in two different Native Language Identification (NLI) scenarios. We compare the results obtained in a traditional NLI document classification task and in a newly introduced sentence classification task, investigating the different role played by the considered features. Finally, we study the impact of a set of selected features extracted from the sentence classifier in document classification.

Italiano. *Partendo da un ampio insieme di caratteristiche linguistiche, presentiamo la prima analisi approfondita del ruolo delle caratteristiche linguistiche nel compito di identificazione della lingua nativa (NLI) in due differenti scenari. Confrontiamo i risultati ottenuti nel tradizionale task di NLI ed in un nuovo compito di classificazione di frasi, studiando il ruolo differente che svolgono le caratteristiche considerate. Infine, studiamo l'impatto di un insieme di caratteristiche estratte dal classificatore di frasi nel task di classificazione di documenti.*

1 Introduction

Native Language Identification (NLI) is the research topic aimed at identifying the native language (L1) of a speaker or a writer based on his/her language production in a non-native language (L2). The leading assumption of NLI research is that speakers with the same L1 exhibit similar linguistic patterns in their L2 productions which can be viewed as traces of the L1 interference phenomena. Thanks to the availability of large-scale benchmark corpora, such as the TOEFL11 corpus (Blanchard et al., 2013), NLI has been recently gaining attention also in the NLP community where it is mainly addressed as a multi-class supervised classification task. This is the approach followed by the more recent systems taking part to the last editions of the NLI Shared Tasks held in 2013 (Tetreault et al., 2013) and 2017 (Malmasi et al., 2017). Typically, these systems exploit a variety of features encoding the linguistic structure of L2 text in terms of e.g. n-grams of characters, words, POS tags, syntactic constructions. Such features are used as input for machine learning algorithms, mostly based on traditional Support Vector Machine (SVM) models. In addition, rather than using the output of a single classifier, the most effective approach relies on ensemble methods based on multiple classifiers (Malmasi and Dras, 2017).

In this paper we want to further contribute to NLI research by focusing the attention on the role played by different types of linguistic features in predicting the native language of L2 writers. Starting from the approach devised by (Cimino and Dell'Orletta, 2017), which obtained the first position in the essay track of the 2017 NLI Shared Task, we carry out a systematic feature selection analysis to identify which features are more effective to capture traces of the native language in L2 writings at sentence and document level.

Our Contributions *(i)* We introduce for the first time a NLI sentence classification scenario, reporting the classification results; *(ii)* We study which features among a wide set of features contribute more to the sentence and to the document classification task; *(iii)* We investigate the contribution of features extracted from the sentence classifier in a stacked sentence-document system.

2 The Classifier and Features

In this work, we built a classifier based on SVM using LIBLINEAR (Rong-En et al., 2008) as ma-

Raw text features
TOEFL11 essay prompt*
Text length (n. of tokens)
Word length (avg. n. of characters)
Average sentence length and standard deviation*
Character n-grams (up to 8)
Word n-grams (up to 4)
Functional word n-grams (up to 3)
Lemma n-grams (up to 4)
Lexical features
Type/token ratio of the first 100, 200, 300, 400 tokens*
Etymological WordNet features (De Melo, 2014)
etymological n-grams (up-to 4)
Morpho–syntactic features
Coarse Part-Of-Speech n-grams (up to 4)
Coarse Part-Of-Speech+Lemma of the following token n-grams (up to 4)
Syntactic features
Dependency type n-grams (sentence linear order) (up to 4)
Dependency type n-grams (syntactic hierarchical order) (up to 4)
Dependency subtrees (dependency of a word + the dependencies to its siblings in the sentence linear order)

Table 1: Features used for document and sentence classification (* only for document).

chine learning library. The set of documents described in Section 3 was automatically POS tagged by the part–of–speech tagger described in (Cimino and Dell'Orletta, 2016) and dependency–parsed by DeSR (Attardi et al., 2009). A wide set of features was considered in the classification of both sentences and documents. As shown in Table 1, they span across multiple levels of linguistic analysis. These features and the classifier were chosen since they were used by the 1st ranked classification system (Cimino and Dell'Orletta, 2017) in the 2017 NLI shared task.

3 Experiments and Results

We carried out two experiments devoted to classify L2 documents and sentences. The training and development set distributed in the 2017 NLI shared task, i.e. the TOEFL11 corpus (Blanchard et al., 2013), was used as training data. It includes 12,100 documents, corresponding to a total of 198,334 sentences. The experiments were tested on the 2017 test set, including 1,100 documents (18,261 sentences).

The obtained macro average F1-scores were: 0.8747 in the document classification task and 0.4035 in the sentence one. As it was expected, the identification of the L1 of the sentences turned out as a more complex task than L1 document classification. Both document and sentence classification

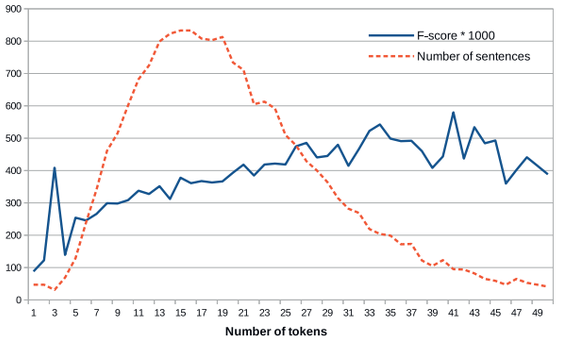


Figure 1: Sentence classification performance across bins of sentences of the same lengths.

are influenced by the number of words but with a different impact. Figure 1 shows that the average performance on sentences is reached for sentences ∼21–token long, which corresponds to the average sentence length for this dataset. As the sentence length increases, the accuracy increases as well. Due to the smaller amount of linguistic evidence, the classification of short sentences is a more complex task. The performance of document classification is more stable: the best f–score is already reached for documents of ∼200–tokens, which corresponds to a very short document compared to the average size of TOEFL11 documents (330 tokens).

Figures 2(a) and 2(b) report the confusion matrices of the two experiments[1]. As it can be seen, both for sentences and documents the best classification performance is obtained for German, Japanese and Chinese, even though with some differences in the relative ranking positions, e.g. German is the top ranked one in the sentence classification scenario and the 2nd ranked one in the document classification one, while Japanese is the best classified L1 in the document experiment and the 4th ranked one in the sentence classification scenario. Conversely, we observe differences with respect to the worst recognized L1s, which are Turkish, Hindi and Korean in the document classification task and Arabic, Spanish and Turkish in the sentence classification one. The two confusion matrices also reveal a peculiar error distribution trend: the confusion matrix of the sentence classification model is much more sparse than the

[1]Since the number of documents and sentences in the two experiments is different, in order to make comparable the values of the two confusion matrices, the sentence classification values were normalized to 100.

document classification one. This means that for each considered L1, the errors made by the sentence classifier are quite similarly distributed over all possible L1s; instead, errors in the document classification scenario are much more prototypical, i.e. the wrong predicted label is assigned to only one or two L1 candidates, which change according to the specific L1. This is shown e.g. by languages belonging to same language family such as Japanese and Korean which belong to the same Altaic family. Specifically, in the document classification scenario Korean is mainly confused with Japanese (10% of errors). This trend holds also in the sentence classification experiment where 17.8% of errors were due to the confusion of Korean with Japanese and vice versa (18.2% of errors). Interestingly enough, the most prototypical errors were also made when contact languages were concerned. This is for example the case of Hindi and Telugu: Hindi documents were mainly confused with Telugu ones (16% of errors) and Telugu documents with Hindi ones (13% of errors). Similarly, in the sentence classification scenario, Hindi sentences were wrongly classified as Telugu sentences in about 20% of cases and vice versa. As previously shown by Cimino et al. (2013), even if these two languages do not belong to the same family, such classification errors might originate from a similar linguistic profile due to language contact phenomena: for instance, both Hindi and Telugu L1 essays are characterized by sentences and words of similar length, or they share similar syntactic structures such e.g. parse trees of similar depth and embedded complement chains governed by a nominal head of similar length.

The behavior of the two classifiers may suggest that *i)* some features could play a different role in the classification of sentences with respect to documents and *ii)* the document classifier can be improved using features extracted from the output of a sentence classifier in a stacked configuration. To investigate these hypotheses, we carried out an extensive feature selection analysis to study the role of the features in the two classification scenarios.

3.1 Feature Selection

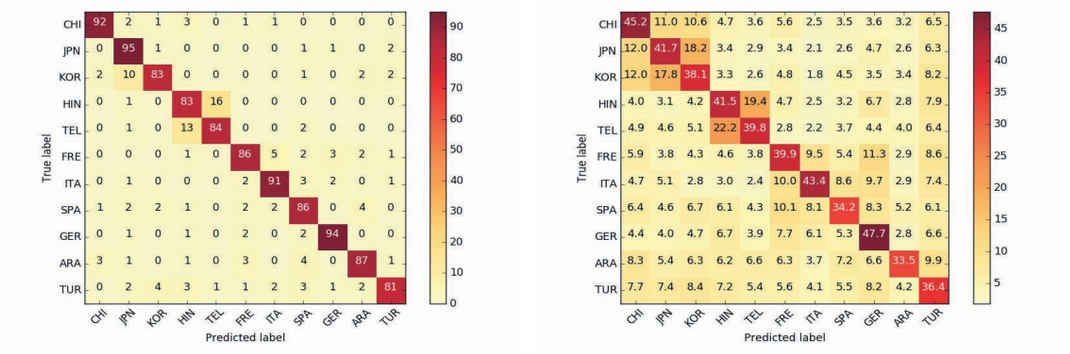
In the first step of the feature selection process, we extracted all the features from the training set and pruned those occurring less than 4 times, obtaining $\sim$ 4,000,000 distinct features both for document and sentence classification. In the second step, we ranked the extracted features through the *Recursive Feature Elimination* (RFE) algorithm implemented in the Scikit-learn library (Pedregosa et al., 2011) using Linear SVM as estimator algorithm. We dropped 1% of features in each iteration. At the end of this step we selected the top ranked features corresponding to $\sim$ 40,000 features both for the sentence and document tasks. These features were further re-ranked using the RFE algorithm (dropping 100 features at each iteration) to allow a more fine grained analysis.

Figure 3(a) compares the percentage of different types of features used in the classification of documents and sentences. As it can be noted, the document classifier uses more words n-grams, especially n-grams characters. Instead, morpho–syntactic and syntactic features are more effective for sentence classification, and the n-grams of lemmas even more than 4 times. Figures 3(b), 3(c) and 3(d) show the variation of relevance of the 40k raw text, morpho-syntactic and syntactic features grouped in bins of 100 features. The lines in the charts correspond to the differences between document and sentence in terms of percentage of a single type of feature in the bin with respect to its total distribution in the whole 40k selected features[2]. Negative values mean that this distribution in the bin is higher for sentence classification.

Among the raw text features (Figure 3(b)), n-grams of words occur more in the 1st bins of document classification, while n-grams of characters and lemma are more relevant in the 1st bins of sentence classification. The n-grams of coarse parts–of-speech are equally distributed in the two rankings, instead both the n-grams of coarse parts–of-speech followed by a lemma and the n-grams of functional words occur more in the 1st bins of sentence classification (Figure 3(c)). This confirms the key role played by lemma in sentence classification.

For what concerns syntactic information (Figure 3(d)), the features that properly capture sentence structure (dependency subtree and the hierarchical syntactic dependencies) are all contained in the first bins of document classification even if their total distribution is lower than in the sentence. This shows that syntactic information is very relevant also when longer texts are classified and that this kind of information is not captured by

[2]Spline interpolation applied for readability purpose.



(a) (b)

Figure 2: Confusion matrix of document (a) and sentence classification (b).

n-grams of words. Feature types with low number of instances are not reported in these charts. Among these, *etymological n-grams* appears in the first bins both for sentence and document, confirming the relevance of the etymological information already proven for NLI document classification (Nastase and Strapparava, 2017). For sentence classification, it is also relevant *sentence length* and *word length*. Instead, for document, *type/token ratio* plays a very important role. Interestingly, the *average sentence length* does not appear in the 40k features; we found instead the *sentence length standard deviation*, showing that what counts more is the variation in length rather than the average value. Even though not contained in the first bins, also *word and document lengths* and the *TOEFL11 essay prompt* are in the top 40k features.

4 Stacked Classifier

The different role of the features in the two L1 classification tasks suggests that we may improve the traditional NLI document classification by combining sentence and document classifiers. We thus evaluated and extended the stacked sentence-document architecture proposed by (Cimino and Dell'Orletta, 2017). In addition to the linguistic features, they proposed a stacked system using the L1 predictions of a pre–trained sentence classifier to train a document classifier. Thus we run several experiments on the NLI Shared Task 2017 test set to assess i) the importance of the sentence classifier in a stacked sentence-document architecture and ii) which features extracted from the predictions of the L1 sentence classifier maximize the accuracy of the stacked system. The sentence classifier assigned a confidence score for each L1 to each sentence of the documents. Based on the confidence score, we defined the following features: for each L1 i) the mean sentence confidence (*avg*), ii) the standard deviation of confidences (*stddev*), iii) the product of the confidences (*prod*), iv) the top–3 highest and lowest confidence values (*top–3 max-min*). The last two features were introduced to mitigate the effect of spike values that may be introduced by considering the max-min L1 confidences used in (Cimino and Dell'Orletta, 2017). The first row of Table 2 reports the result obtained by (Cimino and Dell'Orletta, 2017) by the stacked classifier on the same test set. The second row reports the results of our document system which does not use features extracted from the sentence classifier. The third row reports the result of a classifier that uses only the features extracted from the predictions of the L1 sentence classifier. The following rows report the contribution of each sentence classifier feature in the stacked architecture showing an improvement (with the exception of the product) with respect to the base classifier. The top–3 highest and lowest confidence values are the most helpful features in a stacked architecture. The best result is obtained when using all the sentence classifier features in the base classifier, which is the state-of-the-art on the 2017 NLI test set.

5 Conclusions

We introduced a new NLI scenario focused on sentence classification. Compared to document classification we obtained different results in terms of accuracy and distribution of errors across the L1s. We showed the different role played by a

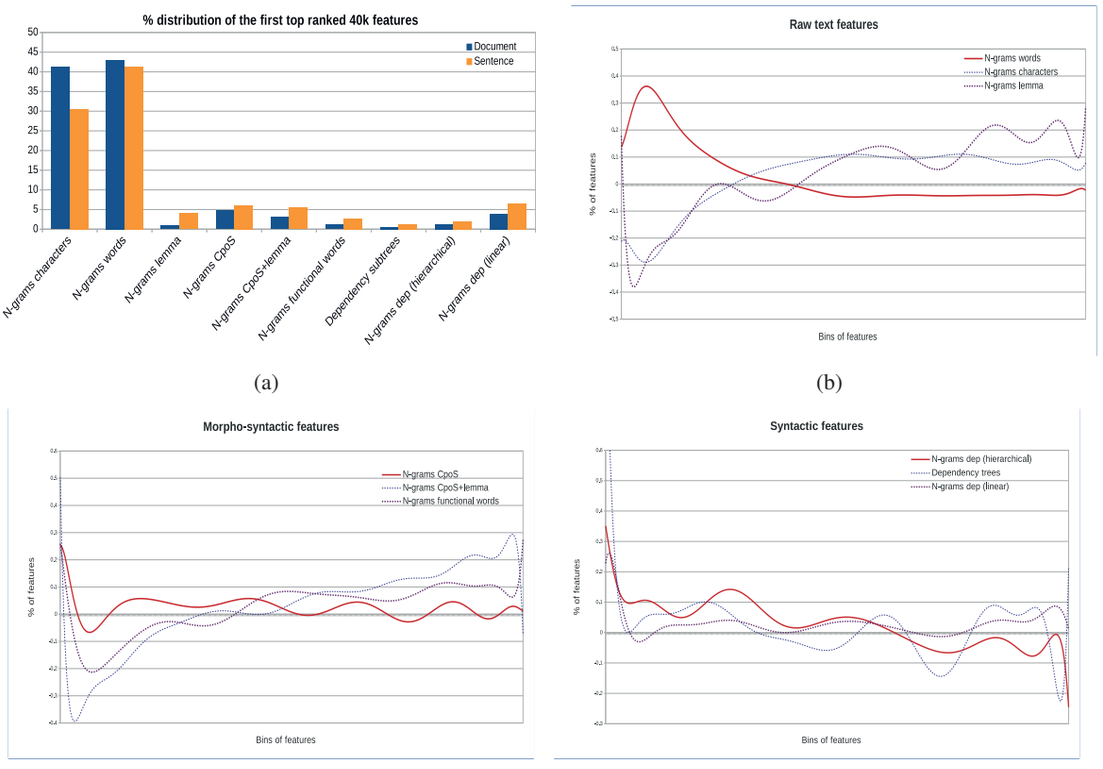


Figure 3: Distribution of the first top ranked 40k features in the document and sentence classification.

Model	F1-Score
Cimino and Dell'Orletta (2017)	0.8818
Base classifier	0.8747
Sentence features	0.8363
Base class. + avg	0.8773
Base class. + stddev	0.8773
Base class. + prod	0.8747
Base class. + top–3 max-min	0.8800
Base class. + all sentence feat.	**0.8828**

Table 2: Results of the stacked system.

wide set of linguistic features in the two NLI scenarios. These differences may justify the performance boost we achieved with a stacked sentence-document system. We also assessed which features extracted from the sentence classifier maximizes NLI document classification.

6 Aknowledgments

The work presented in this paper was partially supported by the 2–year project (2018-2020) SchoolChain – Soluzioni innovative per la creazione, la certificazione, il riuso e la condivisione di unità didattiche digitali all'interno del sistema Scuola, funded by Regione Toscana (BANDO POR FESR 2014-2020).

References

Giuseppe Attardi, Felice Dell'Orletta, Maria Simi, and Joseph Turian. 2009. *Accurate dependency parsing with a stacked multilayer perceptron.* In Proceedings of the 2nd Workshop of Evalita 2009. December, Reggio Emilia, Italy.

Daniel Blanchard, Joel Tetreault, Derrick Higgins, Aoife Cahill, and Martin Chodorow. 2013. *TOEFL11: A Corpus of Non-Native English.* Technical report, Educational Testing Service.

Andrea Cimino and Felice Dell'Orletta. 2016. *Building the state-of-the-art in POS tagging of italian tweets.* In 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), December 5-7.

Andrea Cimino and Felice Dell'Orletta. 2017. *Stacked sentence–document classifier approach for improving native language identification.* In Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications, BEA@EMNLP 2017, September 8, pages 430-437.

Andrea Cimino, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2013. *Linguistic Profiling based on General-purpose Features*

and Native Language Identification. In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, BEA@NAACL-HLT 2013, June 13, 2013, Atlanta, Georgia, USA, pages 207–215.

Shervin Malmasi and Mark Dras. 2017. *Native Language Identification using Stacked Generalization.* arXiv preprint arXiv:1703.06541.

Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano and Yao Qian 2017. *A Report on the 2017 Native Language Identification Shared Task.* In Proceedings of the 12th Workshop on Building Educational Applications Using NLP. BEA@EMNLP 2017, September 8.

Gerard de Melo. 2014. *Etymological Wordnet: Tracing the history of words.* In Proceedings of the 9th Language Resources and Evaluation Conference (LREC 2014), Paris, France. ELRA.

Vivi Nastase and Carlo Strapparava. 2017. *Word etymology as native language interference.* In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2702–2707.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. *Scikit-learn: Machine learning in Python.* Journal of Machine Learning Research, 12:28252830.

Fan Rong-En, Chang Kai-Wei, Hsieh Cho-Jui, Wang Xiang-Rui, and Lin Chih-Jen. 2008. LIBLINEAR: A library for large linear classification. 2008. *LIBLINEAR: A library for large linear classification.* Journal of Machine Learning Research, 9:18711874.

Joel Tetreault, Daniel Blanchard, and Aoife Cahill. 2013. *A Report on the First Native Language Identification Shared Task.* In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, Atlanta, GA, USA. Association for Computational Linguistics.

Gender and Genre Linguistic profiling: a case study on female and male journalistic and diary prose

Eleonora Cocciu[•]
Dominique Brunato[◇], **Giulia Venturi**[◇], **Felice Dell'Orletta**[◇]
[•]Università di Pisa
eleonoracocciu.95@gmail.com
[◇]Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC-CNR)
ItaliaNLP Lab - *www.italianlp.it*
{nome.cognome}@ilc.cnr.it

Abstract

English. This paper intends to investigate the linguistic profile of male- and female-authored texts belonging to two very different textual genres: newspaper articles and diary prose. By using a wide set of linguistic features automatically extracted from text and spanning across different levels of linguistic description, from lexicon to syntax, our analysis highlights the peculiarities of the two examined genres and how the genre dimension is influenced by variation depending on author's gender (and vice versa).

Italiano. *Questo lavoro nasce con lo scopo di definire il profilo linguistico di testi scritti da uomini e da donne appartenenti a due generi testuali molto diversi: la prosa giornalistica e le pagine di diario. Attraverso lo studio di una ampia gamma di caratteristiche linguistiche estratte automaticamente dai testi e riguardanti diversi livelli di descrizione linguistica, che vanno dall'analisi lessicale del testo a quella sintattica, questo lavoro mette in luce le peculiarità dei due generi testuali presi in esame e come la dimensione del dominio dei testi venga influenzata dalla dimensione del genere uomo/donna (e viceversa).*

1 Introduction

Authorship profiling is the task of identifying the author of a given text by defining an appropriate characterization of documents that captures the writing style of authors. It is a well-studied area with applications in various fields, such as intelligence and security, forensics, marketing etc. Over the last years, progress in different disciplines such as Artificial Intelligence, Linguistics and Natural Language Processing (NLP) stimulates new research directions in this field leading to the development of 'computational sociolinguistics', a multidisciplinary field whose goal is to study the relationship between language and social groups using computational methods (Nguyen et al., 2016). With this respect, a particular attention has been paid to the influence of gender as a demographic variable on language use. This is a topic that has attracted linguistic research for decades (see e.g. (Lakoff, 1973)) and has received a renewed interest in recent years in the NLP community. The investigation of possible differences between men's and women's linguistic styles has been carried out by using multivariate analyses taking into account gender-preferential stylistic features (Herring and Paolillo, 2006) and machine learning techniques inferring language models that differ at the level of linguistic patterns learned (e.g. based on n-grams of characters, on lexicon, etc.) (Argamon et al., 2003; Sarawgi et al., 2016). These studies have also moved the interest towards the analysis of possible effects driven by textual genres and topics on gender-specific language preferences. With this respect, in the context of the annual PAN evaluation campaign organized since 2013[1], a cross-genre gender identification shared task was newly introduced (Rangel et al., 2016) in 2016, where participants were asked to predict author's gender with respect to a textual typology different from the one used in training. This scenario turned out to be much more challenging for state-of-the art systems, suggesting that females and males can possibly use a different writing style according to genre. While the cross-genre gender prediction task has received attention for many languages, e.g. English, Portuguese, Arabic, the Italian language will be addressed for the first time by the GxG (Gender X-Genre) shared task in the context

[1]https://pan.webis.de/index.html

of the 2018 EVALITA campaign[2].

In line with this interest in the international community, this paper presents a study on gender variation in writing styles with the aim of investigating if there are gender-specific characteristics that are constant across different genres. We define a methodology to carry out an in-depth linguistic analysis to detect differences and similarities in female- and male-authored writings belonging to two different genres. Similarly to the early work by Argamon et al. (2003) for English, our focus is on the linguistic phenomena that contribute to model men's and women's writings in a cross-genre perspective. The main novelty of this work is that we rely on a very wide set of linguistic features automatically extracted from text and capturing lexical, morpho-syntactic and syntactic phenomena. We choose not to focus our analysis on computer-mediated communication texts, which are more typically used in this context, but on two traditional textual genres, i.e. newspaper articles and diary prose.

2 Corpus Collection

The comparative investigation was carried out on two collection of texts, equally divided by gender, and selected to be representative of two different genres: journalistic prose and diary pages.

	Diaries		**Newspapers**	
	Tokens	Document	Tokens	Document
Women	45,155	100	62,469	100
Men	35,493	100	66,860	100
TOTAL	**80,648**	**200**	**129,329**	**200**

Table 1: Corpus internal composition.

For the journalistic genre we collected 200 documents through the advanced search engine available on the website of *La Repubblica*.

For the second textual genre, we collected 200 texts from the website of the Fondazione Archivio Diaristico Nazionale (*National Diaristic Archive Foundation*). In 1984, the Foundation (which is located in Pieve Santo Stefano in the province of Arezzo (Tuscany)) founded a first public archive containing writings of ordinary people, which was changed into the *National Diaristic Archive Foundation* in 1991. Since 2009 the documentary heritage of the archive has been included in the Code of Cultural Heritage of the State.

[2]https://sites.google.com/view/gxg2018

All selected texts were automatically tagged by the part-of-speech tagger described in (Dell'Orletta, 2009) and dependency parsed by the DeSR parser described in (Attardi et al., 2009). Based on the multi–level output of linguistic annotation, we automatically extracted a wide set of more than 170 linguistic features described in the following section.

3 Linguistic Features

Our approach relies on multi-level linguistic features, which were extracted from the corpus morpho-syntactically tagged and dependency-parsed. They range across different levels of linguistic description and they qualify lexical and grammatical characteristics of a text. These features are typically used in studies focusing on the "form" of a text, e.g. on issues of genre, style, authorship or readability (see e.g. (Biber and Conrad, 2009; Collins-Thompson, 2014; Cimino et al., 2013; Dell'Orletta et al., 2014)).

Raw Text Features: *Token Length* and *Sentence Length* (features 1 and 2 in Table 2): calculated as the average number of characters per tokens and of tokens per sentences.

Number of sentences (feature 3): calculated as the number of sentences of a document.

Lexical Features: *Basic Italian Vocabulary rate features*, all calculated both in terms of lemmata (*L*) and token (*f*), referring to *a)* the internal composition of the vocabulary of the text; we took as a reference resource the Basic Italian Vocabulary by De Mauro (2000), including a list of 7000 words highly familiar to native speakers of Italian (feature 4), and *b)* the internal distribution of the occurring basic Italian vocabulary words into the usage classification classes of 'fundamental words', i.e. very frequent words (feature 5), 'high usage words', i.e. frequent words (feature 6) and 'high availability words', i.e. relatively lower frequency words referring to everyday life (feature 7).

Type/Token Ratio: this feature refers to the ratio between the number of lexical types and the number of tokens. Due to its sensitivity to sample size, this feature is computed for text samples of equivalent length, i.e. the first 100 and 200 tokens (feature 8).

Morpho-syntactic Features *Language Model probability of Part-Of-Speech unigrams*: this feature refers to the distribution of unigram

Part-of-Speech (feature 9).
Lexical density: this feature refers to the ratio of content words (verbs, nouns, adjectives and adverbs) to the total number of lexical tokens in a text.
Verbal morphology: this feature refers to the distribution of verbs (both main and auxiliary) according to their grammatical person, tense and mood (feature 10).
Syntactic Features *Unconditional probability of dependency types*: this feature refers to the distribution of dependency relations (feature 11).
Subordination features: these features (feature 12) include a) the distribution of subordinate vs main clauses and their average length, b) their relative ordering with respect to the main clause, c) the average depth of 'chains' of embedded subordinate clauses and d) the probability distribution of embedded subordinate clauses 'chains' by depth.
Parse tree depth features: this set of features captures different aspects of the parse tree depth and includes the following measures: a) the depth of the whole parse tree, calculated in terms of the longest path from the root of the dependency tree to some leaf (feature 13); b) the average depth of embedded complement 'chains' governed by a nominal head and including either prepositional complements or nominal and adjectival modifiers and their distribution of embedded complement 'chains' by depth (feature 14).
Verbal predicates features: this set of features ranges from the number of verbal roots with respect to number of all sentence roots occurring in a text to their arity. The arity of verbal predicates is calculated as the number of instantiated dependency links sharing the same verbal head.
Length of dependency links: the length is measured in terms of the words occurring between the syntactic head and the dependent (feature 15).

4 Data Analysis

For each considered features we calculated the average value and their standard deviation. To investigate which features characterize male vs. female writings, and the possible influence of genre, we assessed the statistical significance of their variation comparing i) male and female writings, independently from the textual genre and ii) diaries and newspaper articles written by women and men. Table 2 reports features that resulted to vary significantly for at least one of the comparisons we considered. In the second and third columns, headed with *Gender*, it is marked the variation with respect to the textual genre, independently from gender's author, the forth and fifth columns, headed with *Genre*, show the statistical significance of variations with respect to gender.

As it can be seen, the number of features that significantly vary is higher in diaries than in newspaper articles (i.e. 23 vs 11); this may suggest that newspapers are characterized by a quite codified writing style with few variations between female and male authors. When we focus on gender, the effect of genre is more prominent for women, as suggested by the greater number of features (i.e. 35) that significantly varies between female diaries and newspaper articles.

Independently from gender, newspapers are characterized by longer words and, among the considered parts-of-speech, by a higher occurrence of prepositions (both simple and articulated), of nouns and proper nouns, as well as by a more extensive use of punctuation. The nominal style characterizing this genre and suggested by the higher proportion of nouns comes out clearly at syntactic level: newspapers articles greatly differ from diary pages since they present a higher percentage of complements modifying a nouns ([11] Compl. and [11] Prep.) also organized in longer embedded chains ([14]), two features which are more common in highly informative texts than in narrative texts like diaries (Biber and Conrad, 2009). According to the literature, these syntactic structures are typically related to sentence complexity as well as deep syntactic trees ([13]) and long clauses ([12] Avg.len.). These phenomena especially distinguish newspaper articles written by men.

As expected, the language of diaries is identified by features typically characterizing narrative texts: the considered collection contains longer sentences, especially male diaries, and a lower percentage of high usage ([6] (f)) and high availability ([7] (f)) lexicon belonging to the *Basic Italian Vocabulary* (BIV). Features capturing the verbal morphology reflect the narrative style used to refer to experiences occurred in the past: the diaries (especially those by male authors) contain a higher usage of imperfect tense and more auxiliary verbs, possibly composing past tenses. In addition, a number of features suggests that the diary

Feature	Gender		Genre		Diaries				Newspaper articles			
	D	J	W	M	Women		Men		Women		Men	
Raw text features												
[1]	-	⋆⋆⋆	⋆	⋆⋆⋆	4.64	(0.31)	4.81	(0.25)	5.07	(0.23)	5.2	(0.22)
[2]	⋆	-	-	⋆	23.95	(20.74)	25.40	(14.53)	25.43	(6.78)	25.49	(6.36)
[3]	-	-	⋆⋆⋆	-	22.16	(14.75)	21.9	(15.61)	26.6	(12.33)	27.8	(11.36)
Lexical features												
[4] (L)	-	-	⋆⋆⋆	-	78.6	(5.44)	72.3	(10.2)	69	(5.47)	68.1	(4.93)
[4] (f)	-	-	⋆⋆⋆	-	88.8	(4.07)	83.9	(6.91)	81.5	(4.00)	80.7	(3.8)
[5] (L)	-	-	⋆⋆⋆	-	83.7	(4.16)	80.2	(4.39)	76.8	(4.14)	76.6	(3.63)
[5] (f)	-	-	⋆⋆⋆	-	81.4	(3.58)	78.9	(3.98)	74.4	(3.93)	74.1	(3.55)
[6] (L)	-	-	⋆⋆⋆	-	11.8	(3.91)	15	(3.84)	17.8	(3.65)	18.3	(3.33)
[6] (f)	⋆⋆⋆	-	-	-	11	(2.52)	12.4	(3.02)	13.9	(2.50)	14.1	(2.36)
[7] (L)	-	-	⋆⋆⋆	-	4.48	(1.85)	4.75	(1.70)	5.42	(1.83)	5.06	(1.68)
[7] (f)	⋆⋆⋆	-	⋆⋆⋆	-	7.55	(2.22)	8.67	(2.53)	11.3	(2.43)	11.8	(2.41)
[8] 100 (f)	-	-	⋆	⋆	0.83	(0.05)	0.83	(0.06)	0.85	(0.05)	0.85	(0.05)
[8] 200 (L)	-	-	⋆	-	0.60	(0.05)	0.61	(0.05)	0.62	(0.04)	0.63	(0.04)
[8] 200 (f)	-	-	⋆⋆⋆	⋆	0.72	(0.05)	0.73	(0.05)	0.75	(0.04)	0.75	(0.04)
Morpho-syntactic features												
[9] Prep.	⋆	⋆⋆⋆	⋆⋆⋆	⋆	11.5	(2.68)	12.6	(2.90)	15.22	(2.12)	16.19	(1.91)
[9] Artic.prep.	⋆	⋆⋆⋆	⋆	⋆⋆⋆	3.27	(1.82)	3.91	(1.53)	5.76	(1.69)	6.50	(1.44)
[9] Pron.	-	-	⋆⋆⋆	⋆	8	(2.79)	7.41	(2.64)	4.37	(1.57)	4.26	(1.21)
[9] Punct.	-	⋆⋆⋆	-	-	13.5	(3.45)	12.6	(3.35)	13.66	(2.42)	12.48	(2.09)
[9] Aux.verb.	⋆⋆⋆	-	-	⋆	2.38	(1.38)	1.80	(1.28)	2.18	(1.52)	2.03	(0.96)
[9] Adj.	-	-	⋆	⋆⋆⋆	4.86	(1.80)	4.89	(1.75)	5.26	(1.58)	5.70	(1.72)
[9] Poss.adj.	⋆	-	-	-	1.46	(0.99)	1.06	(0.86)	0.56	(0.50)	0.60	(0.41)
[9] Neg.adv.	⋆⋆⋆	-	-	⋆⋆⋆	1.68	(1.08)	1.14	(0.65)	0.94	(0.58)	0.85	(0.46)
[9] Subord.conj.	⋆	-	-	-	1.64	(0.92)	1.45	(0.93)	0.95	(0.66)	0.99	(0.54)
[9] Nouns	-	-	⋆⋆⋆	-	19.5	(3.77)	22.8	(4.57)	26.67	(3.36)	26.99	(2.73)
[9] Prop.nouns	⋆	-	⋆⋆⋆	-	2.64	(1.68)	3.70	(3.05)	6.42	(3.11)	6.71	(2.71)
[10] 1p.plur.	⋆	-	-	⋆	4.01	(6.16)	5.35	(8.21)	3.87	(4,74)	2.62	(4.31)
[10] 3p.plur.	-	-	⋆	⋆	14.5	(10.52)	15.5	(12.96)	18.04	(9.17)	18.45	(9.98)
[10] 1p.sing.	⋆	-	⋆	-	20.9	(13.40)	14.5	(12.97)	3.19	(4.41)	2.95	(5.05)
[10] 2p.sing.	-	-	⋆	-	2.80	(5.27)	1.80	(3.45)	0.69	(1.30)	0.45	(1.13)
[10] 3p.sing.	⋆	-	-	⋆	38	(13.28)	45.2	(16.34)	49.64	(13)	50.33	(12.49)
[10] 3p.plur.	-	-	⋆⋆⋆	-	2.31	(3.21)	2.75	(4.50)	6.01	(6.38)	6.34	(5.66)
[10] 1p.sing.	⋆	-	⋆	⋆	7.26	(7.60)	4.32	(6.03)	1.8	(3.91)	0.75	(1.73)
[10] Future	-	-	-	⋆	5.59	(7.40)	2.98	(5.04)	5.94	(8.08)	6.79	(8.95)
[10] Imperfect	⋆	-	⋆⋆⋆	-	21.9	(24.48)	26.2	(24.01)	8.61	(9.10)	9.14	(11.40)
[10] Past	-	-	⋆	-	8.78	(15.17)	9.74	(14.88)	1.51	(4.81)	2.37	(4.70)
Syntactic features												
[11] Compl.	⋆⋆⋆	⋆⋆⋆	⋆⋆⋆	-	8.80	(2.15)	9.96	(2.55)	12.10	(1.90)	13	(1.82)
[11] Prep.	⋆⋆⋆	⋆⋆⋆	⋆	⋆	11.5	(2.69)	12.7	(2.88)	15.2	(2.12)	16.2	(1.91)
[11] Punct.	⋆	⋆	⋆	⋆⋆⋆	11.4	(3.05)	10.2	(3)	12.3	(2.22)	11.4	(1.96)
[11] Temp.mod.	⋆	-	⋆⋆⋆	-	0.89	(0.69)	0.61	(0.57)	0.57	(0.43)	0.51	(0.37)
[11] Pred.comp.	⋆	-	-	⋆⋆⋆	2.46	(1.03)	2.03	(1.04)	1.68	(0.70)	1.55	(0.60)
[11] Aux.	⋆	-	-	⋆	2.30	(1.36)	1.72	(1.29)	2.11	(1.56)	1.97	(0.97)
[12] Main	-	-	⋆	⋆⋆⋆	61.1	(14.8)	61.8	(13.7)	67.5	(10.3)	68.1	(10.13)
[12] Sub.	-	-	⋆	⋆⋆⋆	38.9	(14.8)	38.2	(13.7)	32.5	(10.3)	31.9	(10.13)
[12] Avg.len.	⋆⋆⋆	⋆	⋆	-	7.19	(1.17)	7.98	(1.72)	9.20	(1.57)	9.56	(1.46)
[12] (post-verb)	-	⋆	-	-	90.1	(16.9)	87.4	(21.8)	84.2	(13.9)	88.9	(11.06)
[12] (pre-verb)	-	-	⋆⋆⋆	⋆	7.88	(11)	9.56	(15.5)	15.8	(13.9)	11	(11.06)
[13]	⋆	⋆	-	⋆	5.61	(2.84)	6.34	(2.55)	6.21	(1.22)	6.60	(1.18)
[14]	-	⋆	-	-	1.17	(0.12)	1.19	(0.11)	1.29	(0.11)	1.31	(0.08)
[14] (len 3)	-	-	⋆	⋆	1.72	(3.69)	1.68	(2.52)	3.84	(3.14)	3.75	(2.35)
[15]	-	-	⋆⋆⋆	⋆	9.12	(7.47)	9.56	(4.87)	9.84	(2.65)	9.95	(2.66)

Table 2: ⋆⋆⋆ highly statistically significant ($p < 0.001$), ⋆ statistically significant ($p < 0.05$), - any statistically significant features characterizing the two considered textual genres (column *Gender*), i.e. diaries (*D*) vs. newspaper articles (*J*) independently from gender; the two genders (column *Genre*), i.e. women (*W*) vs. men (*M*) independently from textual genre; average feature values and standard deviation in parenthesis for the four different sub-corpora. Features $[1-3]$, [12] Avg.len, [13], [14], [15] are absolute values, the others are percentage distributions.

prose is typically characterized by a more subjective writing style. Namely, the collected diaries present a more extensive use of the first and second singular person verbs, especially those written by women (i.e. 1st person verb: 20.9 women vs 14.5 men), and a higher distribution of possessive adjectives.

If we focus on the gender dimension, our results show that female writings are characterized by features typically found in easier-to-read texts, according to the literature on readability assessment (Collins-Thompson, 2014). This is especially true for the following parameters: they contain shorter words, more fundamental lexicon ([5] (L), (f)), less high usage ([6] (L), (f)) and high availability ([7] (L), (f)) lexicon. At syntactic level, sentences written by women are also characterized by shorter clauses, shorter dependency links and less shallow syntactic trees, as well as by a more canonical use of subordinate clauses in pre-verbal position. On the contrary, men diaries share more features of linguistic complexity: they contain longer sentences, more complex lexicon, a higher percentage of nouns and proper nouns and syntactic features typically occurring in complex structures.

5 Conclusion

We have presented a cross-genre linguistic profiling investigation comparing male and female texts in Italian. We examined a large set of linguistic features, intercepting lexical and syntactic phenomena, which were extracted from two very different textual genres: newspaper articles and diary prose. As expected, the comparative analysis highlighted a number of differences between the two genres, due to the more subjective language characterizing diaries with respect to journalistic prose. Interestingly, we also highlighted that some linguistic features characterize gender dimension and, even more interestingly, we found statistically significant variations also in an objective prose such as newspaper articles.

6 Acknowledgements

The work reported in the paper was partially supported by the 2–year project (2017-2019) UBIMOL, UBIquitous Massive Open Learning, funded by Regione Toscana (BANDO POR FESR 2014-2020).

References

S. Argamon, M. Koppel, J. Fine, and A. Shimoni 2003. Gender, Genre, and Writing Style in Formal Written Texts. *Text*, 23(3).

G. Attardi, F. Dell'Orletta, M. Simi, and J. Turian. 2009. Accurate dependency parsing with a stacked multilayer perceptron. In *Proceedings of EVALITA 2009 - Evaluation of NLP and Speech Tools for Italian 2009*, Reggio Emilia, Italy, December 2009.

D. Biber and S. Conrad. 2009. Genre, Register, Style. Cambridge: CUP.

A. Cimino, F. Dell'Orletta, G. Venturi, and S. Montemagni. 2013. Linguistic Profiling based on General-purpose Features and Native Language Identification. *Proceedings of Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, Atlanta, Georgia, June 13, pp. 207-215.

K. Collins-Thompson. 2014. Computational Assessment of text readability. *Recent Advances in Automatic Readability Assessment and Text Simplification. Special issue of International Journal of Applied Linguistics*, 165:2, John Benjamins Publishing Company, 97-135.

F. Dell'Orletta. 2009. Ensemble system for part-of-speech tagging. In *Proceedings of EVALITA 2009 - Evaluation of NLP and Speech Tools for Italian 2009*, Reggio Emilia, Italy, December 2009.

F. Dell'Orletta, M. Wieling, A. Cimino, G. Venturi, and S. Montemagni. 2014. Assessing the Readability of Sentences: Which Corpora and Features. *Proceedings of the 9th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2014)*, Baltimore, Maryland, USA.

T. De Mauro. 2000. *Grande dizionario italiano dell'uso (GRADIT)*. Torino, UTET.

S. C. Herring and J. C. Paolillo. 2006. Gender and genre variation in weblogs. *Journal of Sociolinguistics*, 10/4, pp. 439–459.

R. T. Lakoff. 1973. Language and woman's place. In Language in Society, 2/1, pp. 45–80.

D. Nguyen, A.S. Doruz, C.P. Ros, and F.M.G. de Jong. 2016. Computational Sociolinguistics: A Survey. *Computational Linguistics*, Vol. 42, No. 3, Pages 537-593.

F. Rangel, P. Rosso, B. Verhoeven, W. Daelemans, M. Pottast, and B. Stein. 2016. Overview of the 4th Author Profiling Task at PAN 2016: Cross-Genre Evaluations. In: Balog K., Capellato L., Ferro N., Macdonald C. (Eds.) *CLEF 2016 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings*. CEUR-WS.org, vol. 1609, pp. 750-784.

R. Sarawgi, K. Gajulapalli, and Y. Choi. 2011. Gender Attribution: Tracing Stylometric Evidence Beyond Topic and Genre. *Proceedings of the Fifteenth Conference on Computational Natural Language Learning (CoNLL 2011)*, Portland, Oregon, USA, June 23-24, 2011, pp. 78–86.

Conceptual Abstractness: from Nouns to Verbs

Davide Colla, Enrico Mensa, Aureliano Porporato, Daniele P. Radicioni
Dipartimento di Informatica – Università degli Studi di Torino
firstname.surname@unito.it

Abstract

English. Investigating lexical access, representation and processing involves dealing with conceptual abstractness: abstract concepts are known to be more quickly and easily delivered in human communications than abstract meanings (Binder et al., 2005). Although these aspects have long been left unexplored, they are relevant: abstract terms are widespread in ordinary language, as they contribute to the realisation of various sorts of figurative language (metaphors, metonymies, hyperboles, *etc.*). Abstractness is therefore an issue for computational linguistics, as well. In this paper we illustrate how to characterise verbs with abstractness information. We provide an experimental evaluation of the presented approach on the largest existing *corpus* annotated with abstraction scores: our results exhibit good correlation with human ratings, and point out some open issues that will be addressed in future work.

Italiano. *In questo lavoro presentiamo il tema dell'astrattezza come una caratteristica diffusa del linguaggio, e un nodo cruciale nell'elaborazione automatica del linguaggio. In particolare illustriamo un metodo per la stima dell'astrattezza che caratterizza i verbi a partire dalla composizione dei punteggi di astrattezza degli argomenti dei verbi utilizzando la risorsa* Abs-COVER.

1 Introduction

Surprisingly enough, most of frequently used words (70% of the top 500) seem to be associated to abstract concepts (Recchia and Jones, 2012). Coping with abstractness is thus central to the investigation of lexical access, representation, and processing and, consequently, to build systems dealing with natural language. Information on conceptual abstractness impacts on many diverse NLP areas, such as word sense disambiguation (WSD) (Kwong, 2008), the semantic processing of figurative uses of language (Turney et al., 2011; Neuman et al., 2013), automatic translation and simplification (Zhu et al., 2010), the processing of social tagging information (Benz et al., 2011), and many others, as well. In the WSD task, abstractness has been investigated as a core feature in the fine tuning of WSD algorithms (Kwong, 2007): in particular, experiments have been carried out showing that "words toward the concrete side tend to be better disambiguated that those lying in the mid range, which are in turn better disambiguated than those on the abstract end" (Kwong, 2008).

A recent, inspiring, special issue hosted by the Topics in Cognitive Science journal on 'Abstract Concepts: Structure, Processing, and Modeling' provides various pointers to tackle abstractness, by posing it as a relevant issue for several disciplines such as psychology, neuroscience, philosophy, general AI and, of course, computational linguistics (Bolognesi and Steen, 2018). As pointed out by the Editors of the special issue, the investigation on abstract concepts is central in the multidisciplinary debate between grounded views of cognition *versus* modal (or symbolic) views of cognition. In short, cognition might be *embodied* and grounded in perception and action (Gibbs Jr, 2005): accessing concepts would amount to retrieving and instantiating perceptual and motoric experience. Typically, abstract concepts, that have no direct counterpart in terms of perceptual and motoric experience, are accounted for by such theories with difficulty. On the other side, modal approaches to concepts are mostly in the realm of distributional semantic models: in this view, the

meaning of *rose* is "the product of statistical computations from associations between *rose* and concepts like *flower*, *red*, *thorny*, and *love*" (Louwerse, 2011).[1]

While we do not enter this passionate debate, we start by considering that distributional models are of little help in investigating abstractness, with some notable exceptions, such as the interesting links between abstractness and emotional content drawn in (Lenci et al., 2018). In fact, whilst distributional models can be easily used to express similarity and analogy (Turney, 2006), since they are basically built on co-occurrence matrices, they are largely acknowledged to convey vague associations rather than defining a semantically structured space (Lenci, 2018). As illustrated in the following, our approach is different from such mainstream approach, in that the conceptual descriptions used to compute abstractness and contained in the lexical resources COVER (Mensa et al., 2018c) and ABS-COVER (Mensa et al., 2018b)[2] are aimed at putting together the lexicographic precision and richness of BabelNet (Navigli and Ponzetto, 2012) and the common-sense knowledge available in ConceptNet (Havasi et al., 2007).

One preliminary issue is, of course, how to define abstractness, since no general consensus has been reached on what should be measured when considering abstractness or, conversely, concreteness (Iliev and Axelrod, 2017). The term 'abstract' has two main interpretations: *i)* what is *not perceptually salient*, and *ii)* what is *less specific*, and referred to the more general categories contained in the upper levels of a taxonomy/ontology. According to the second view, the concreteness or *specificity* —the opposite of abstractness— can be defined as a function of the distance intervening between a concept and a parent of that concept in the top-level of a taxonomy or ontology (Changizi, 2008): the closer to the root, the more abstract. In this setting, existing taxonomies and ontology-like resources can be directly employed, such as WordNet (Miller et al., 1990) or BabelNet (Navigli and Ponzetto, 2012).

In this work we single out the first aspect, and focus on perceptually salient abstractness; we start from a recent work where we proposed an algorithm to compute abstractness (Mensa et al., 2018a) for concepts contained in COVER (Mensa et al., 2018c; Lieto et al., 2016),[3] and we extend that approach in order to characterise also verbs, whose abstractness is presently computed by combining the abstractness of their (nominal) dependents. Different from most literature we treat abstractness as a feature of word meanings (senses), rather than a feature of word forms (terms).

2 Related Work

Due to space reasons we cannot provide a full account of the related work from a scientific perspective nor about applications and systems; we limit to adding a mention to the closest and most influential approaches. Abstractness has been used to analyse web image queries, and to characterise them in terms of processing difficulty (Xing et al., 2010). In particular, the abstractness associated to nouns is computed by checking the presence of the *physical entity* synset among the hypernyms of senses in the WordNet taxonomy. This approach also involves a disambiguation step, which is performed through a model trained on the SemCor *corpus* (Miller et al., 1993).

Methods based on both (perceptual *vs.* specificity-based) notions of abstractness are compared in (Theijssen et al., 2011). Specifically, the authors of this work report a 0.17 Spearman correlation between scores obtained with the method by (Changizi, 2008) and those obtained by (Xing et al., 2010), in line with the findings about the correlation of values based on the two definitions. This score can be considered as an estimation of the overlap of the two notions of abstractness: the poor correlation seems to suggest that they are rather distinct.

Finally, the abstractness scores by (Xing et al., 2010) and (Changizi, 2008) have been compared with those in the Medical Research Council Psycholinguistic (MRC) Dataset (Coltheart, 1981) reporting, respectively, a 0.60 and 0.29 Spearman correlation with the human ratings.

[1] *Modal* or *symbolic* views of cognition should not be confused with the symbolic AI, based on high-level representations of problems, as outlined by the pioneering work by Newell and Simon (such as, e.g., in (Newell, 1980)), that was concerned with physical symbol systems

[2] `https://ls.di.unito.it`.

[3] COVER is a lexical resource developed in the frame of a long-standing research aimed at combining ontological and common-sense reasoning (Ghignone et al., 2013; Lieto et al., 2015; Lieto et al., 2017).

3 From Nouns to Verbs Abstractness

In this Section we recall the conceptual representation implemented in COVER; we then describe how the resource has evolved into ABS-COVER, that provides nouns with abstractness scores. We then show how abstractness scores are computed for verbs.

COVER is a lexical resource aimed at hosting general conceptual representations. Each concept c is identified through a BabelNet synset ID and described as a vector representation $\vec{c}$, composed by a set of semantic dimensions $\mathcal{D} = \{d_1, d_2, \dots d_n\}$. Each such dimension encodes a relationship like, e.g., ISA, USEDFOR, HASPROPERTY, CAPABLEOF, *etc.* and reports the concepts that are connected to c along the dimension d_i. The vector space dimensions are based on ConceptNet relationships. The dimensions are filled with BabelNet synset IDs, so that finally each concept c in COVER can be defined as

$$\vec{c} = \bigcup_{d \in \mathcal{D}} \{\langle ID_d, \{c_1, \cdots, c_k\}\rangle\}$$

where ID_d is the identifier of the d-th dimension, and $\{c_1, \cdots, c_k\}$ is the set of values (concepts themselves) filling d.

3.1 Annotation of Nouns in ABS-COVER

The annotation of COVER concepts is driven by the hypothesis that the abstractness of a concept can be computed by the abstractness of its ancestor(s) (basically, its hypernyms in WordNet), resorting to their top level super class, either abstract or concrete entity, as previously done in (Xing et al., 2010). In ABS-COVER every concept is automatically annotated with an abstractness score ranging in the $[0, 1]$ interval, where the left bound 0.0 features fully concrete concepts, and the right bound 1.0 stands for maximally abstract concept. The main algorithm consists of two steps, the *base score computation* and the *smoothing phase* (Mensa et al., 2018a).

The **base score computation** is designed to compute a base abstractness score for each element e in COVER. *a*) The algorithm first looks up for the concepts associated to e in BabelNet and retrieves the corresponding set of WordNet hypernyms: if these contain the *physical entity* concept, the base abstractness score of e is set to 0.0; otherwise it is set to 1.0. *b*) In case of failure (i.e., no WordNet synset ID can be found for e), the direct BabelNet hypernyms of e are retrieved and the step *a* is performed for each such hypernyms. Finally, *c*) in case taxonomic information cannot be exploited for e, the BabelNet main gloss for e is retrieved and disambiguated, thus obtaining a set of concepts N. We then perform steps (*a* and *b*) for each noun $n \in N$. The gloss scores are averaged and the result is assigned as score of e. If the function fails in all of these steps, the abstractness score is set to -1, indicating that no suitable score could be computed. For example, the concept *bomb* as "an explosive device fused to explode under specific conditions",[4] is connected to *physical entity* through its hypernyms in WordNet; thus, its base score is set to 0.0.

The **smoothing phase** focuses on the tuning of the base scores previously obtained by following human perception accounts; to do so, we employ the common-sense knowledge available in COVER. Given a vector $\vec{c}$ in the resource, we explore a subset of its dimensions:[5] all the base abstractness scores of the concepts that are values for these dimensions are retrieved, and the average score $s_{\text{values-avg}}$ is computed. The score $s_{\text{values-avg}}$ is then in turn averaged with $s_{\text{vec-base}}$, that is the base score of $\vec{c}$, thus obtaining the final score for the COVER vector. Continuing our previous example concerning the concept *bomb*, the average abstractness score of its dimension values is mostly low. Specifically, the "bomb" vector in COVER contains, for instance, "bombshell" (with a score of 0.0), "war" (with a score of 1.0) and "explosive material" (with a score of 0.0). The average of *bomb*'s values is 0.2245 and thus the final, smoothed abstractness score for *bomb* is set to 0.112.

3.2 Annotation of Verbs

COVER does not include a conceptual representation for verbs: only nouns are present herein, and this is currently an active line of research aiming at ameliorating the resource. However, in order to build practical applications, we needed to be able to also characterise verb abstractness (Mensa et al., 2018b). In this work we do not aim at extending COVER with verbs representations, but rather to see if the nouns in ABS-COVER can be

[4] Featured by the WordNet synset ID `wn:02866578n`.

[5] We presently consider the following dimensions: RELATEDTO, FORMOF, ISA, SYNONYM, DERIVEDFROM, SIMILARTO and ATLOCATION.

exploited in order to compute verb abstractness.

We start by representing the meaning of verbs in terms of their argument distribution, which is common practice in NLP. We followed this intuition: abstract senses are expected to have more abstract dependents than concrete ones. For example, let us consider the verb *drop*. To drop may be —concretely— intended as "to fall vertically". In this case, it takes concrete nouns as dependents, such as, e.g., in "the bombs are dropping on enemy targets". In a more abstract meaning to drop is "to stop pursuing or acting": in this case its dependents are more abstract nouns, such as, e.g., in "to drop a lawsuit". Although some counterexamples may also be provided, we found that this assumption holds in most cases.

We retrieved the $1,000$ most common verbs from the Corpus of Contemporary American English, which is a corpus covering different genres, such as spoken language, fiction, magazines, newspaper, academic.[6] In order to collect statistics on the argument structure of the considered verbs, we then sampled $3,000$ occurrences of such verbs in the WaCkypedia_EN *corpus*, a 2009 dump of the English Wikipedia, containing about 800 million tokens, tagged with POS, lemma and full dependency parsing (Baroni et al., 2009).[7] All trees containing the verbs along with their dependencies were collected, and such sentences have been passed to the Babelfy API for disambiguation. We retained all verb senses with at least 5 dependents that are present in COVER. The abstractness score of each sense has been computed by averaging the abstractness scores of all its dependents.

4 Evaluation

In order to assess the computed abstractness scores we make use of the Brysbaert Dataset, which is to date the largest corpus of English terms annotated with abstractness scores. It has been acquired through crowdsourcing, and it contains $39,945$ annotated terms (Brysbaert et al., 2014). One chief issue clearly stems from the fact that the human abstractness ratings are referred to terms rather than to senses, which may bias the results of comparisons between the figures used as a ground truth values and the abstractness scores computed by our system. This issue has been experimentally explored in (Mensa et al., 2018a), where different selectional schemes have been tested to pick up a sense from those associated to a given term. The best results, in terms of both Pearson r correlation and of Spearman ρ correlation with human ratings, have been reached by choosing a 'best' sense for the term t based on the distribution of the senses associated to t in the *SemCor* corpus (Miller et al., 1993). Specifically, the correlations between the abstractness scores in ABS-COVER and the human ratings in the Brysbaert Dataset amount to $r = 0.653$ and to $\rho = 0.639$.

	MaxAbs	*MinAbs*	*MaxDep*	*BestSns*
Pearson r	0.4163	0.4581	0.5103	0.4729
Spearman ρ	0.4037	0.4690	0.5117	0.4792

Table 1: Correlation results obtained by comparing our system's abstractness scores against the human ratings in BRYS.

We presently compare the human ratings contained in the Brysbaert *corpus* and the abstractness score associated to one verb sense (corresponding to each lexical entry in the dataset), as computed by our system. We report the correlation scores obtained by selecting the senses based on four strategies:

1. the sense with highest abstractness (*MaxAbs*);
2. the sense with lowest abstractness (*MinAbs*);
3. the sense with the highest number of dependents (*MaxDep*);
4. the sense returned as the best sense through the BabelNet API (*BestSns*).

The obtained results are reported in Table 1. The differences in the scores reported in Table 1 provide tangible evidence that the problem of selecting the correct sense for a verb is a crucial one. E.g., if we consider the verb 'eat', the sense described as "Cause to deteriorate due to the action of water, air, or an acid (example: The acid *corroded* the metal)" and the sense described as "Worry or cause anxiety in a persistent way (What's *eating* you?)" exhibit fully different abstractness characterisation. In order to decouple the assessment of the abstractness scores from that of the sense selection, we randomly selected 400 verbs, and manually associated them with an *a priori* reasonable sense,[8] annotated through the cor-

[6] http://corpus.byu.edu/full-text/.

[7] http://wacky.sslmit.unibo.it/doku.php?id=corpora.

[8] Disambiguation proper would require to select a sense in accordance with a given context.

	FULL-400	Pruning ϑ_1
Pearson r	0.6419	0.6848
Spearman ρ	0.6634	0.6854

Table 2: Correlation scores obtained by manually choosing the main sense for 400 verbs (column FULL-400), and correlation scores obtained by removing from the FULL-400 verbs those with abstractness $\leq .1$ (column ϑ_1 pruning).

responding BabelNet Synset Id. This annotation process is definitely an arbitrary one (only one annotator, thus no inter annotator agreement was recorded, *etc.*), and it should be considered as an approximation to the senses underlying the human ratings available in the Brysbaert *corpus*. The correlation scores significantly raise, as illustrated in the first column of Table 2, thus confirming the centrality of the sense selection step.

Furthermore, we observed that most mismatches in the computation of the abstractness scores occur when the verb is featured by very low (lower than 0.1) abstractness score. To corroborate such intuition, we have then pruned from our data set the verbs whose annotated score is lower than a threshold $\vartheta_1 = 0.1$, finally yielding 383 verbs. In this experimental setting we obtained higher correlation scores, thereby confirming that the computation of more concrete entities needs to be improved, as illustrated in the second column of Table 2.

5 Conclusions

In this paper we have introduced a method to compute verbs abstractness based on the ABS-COVER lexical resource. We reported on the experimentation, and discussed the obtained results, pointing out some issues such as the problem of the sense selection, and the difficulty in characterising more concrete concepts.

As regards as future work, the simple averaging scheme on dependents' abstractness scores can be refined in many ways, e.g., by differentiating the contribution of different sorts of dependents, or based on their distribution. Yet, the set of relations that constitute the backbone of ABS-COVER can be further exploited both for computing the abstractness of dependents, and, in the long term, for generating explanations about the obtained abstractness scores, in virtue of the set of relations at the base of the explanatory power of COVER (Colla et al., 2018). Finally, we plan to explore whether and to what extent our lexical resource can be combined with distributional models, in order to pair those strong associative features with the more semantically structured space described by ABS-COVER.

References

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

Dominik Benz, Christian Körner, Andreas Hotho, Gerd Stumme, and Markus Strohmaier. 2011. One tag to bind them all: Measuring term abstractness in social metadata. In *Extended Semantic Web Conference*, pages 360–374. Springer.

Jeffrey R Binder, Chris F Westbury, Kristen A McKiernan, Edward T Possing, and David A Medler. 2005. Distinct brain systems for processing concrete and abstract concepts. *Journal of cognitive neuroscience*, 17(6):905–917.

Marianna Bolognesi and Gerard Steen. 2018. Editors' introduction: Abstract concepts: Structure, processing, and modeling. *Topics in cognitive science*.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46(3):904–911.

Mark A Changizi. 2008. Economically organized hierarchies in wordnet and the oxford english dictionary. *Cognitive Systems Research*, 9:214–228.

Davide Colla, Enrico Mensa, Daniele P. Radicioni, and Antonio Lieto. 2018. Tell Me Why: Computational Explanation of Conceptual Similarity Judgments. In *Procs. of the 17th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*, Communications in Computer and Information Science (CCIS), Cham. Springer.

Max Coltheart. 1981. The MRC psycholinguistic database. *The Quarterly Journal of Experimental Psychology Section A*, 33(4):497–505.

Leo Ghignone, Antonio Lieto, and Daniele P. Radicioni. 2013. Typicality-Based Inference by Plugging Conceptual Spaces Into Ontologies. In Antonio Lieto and Marco Cruciani, editors, *Proceedings of the International Workshop on Artificial Intelligence and Cognition*. CEUR.

Raymond W Gibbs Jr. 2005. *Embodiment and cognitive science*. Cambridge University Press.

Catherine Havasi, Robert Speer, and Jason Alonso. 2007. ConceptNet: A Lexical Resource for Common Sense Knowledge. *Recent advances in natural language processing V: selected papers from RANLP*, 309:269.

Rumen Iliev and Robert Axelrod. 2017. The paradox of abstraction: Precision versus concreteness. *Journal of psycholinguistic research*, 46(3):715–729.

Oi Yee Kwong. 2007. CITYU-HIF: WSD with human-informed feature preference. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 109–112. Association for Computational Linguistics.

Oi Yee Kwong. 2008. A preliminary study on the impact of lexical concreteness on word sense disambiguation. In *Proceedings of the 22nd Pacific Asia Conference on Language, Information and Computation*.

Alessandro Lenci, Gianluca E Lebani, and Lucia C Passaro. 2018. The emotions of abstract words: A distributional semantic analysis. *Topics in cognitive science*.

Alessandro Lenci. 2018. Distributional models of word meaning. *Annual review of Linguistics*, 4:151–171.

Antonio Lieto, Andrea Minieri, Alberto Piana, and Daniele P. Radicioni. 2015. A knowledge-based system for prototypical reasoning. *Connection Science*, 27(2):137–152.

Antonio Lieto, Enrico Mensa, and Daniele P. Radicioni. 2016. A Resource-Driven Approach for Anchoring Linguistic Resources to Conceptual Spaces. In *Proceedings of the XVth International Conference of the Italian Association for Artificial Intelligence*, volume 10037 of *Lecture Notes in Artificial Intelligence*, pages 435–449. Springer.

Antonio Lieto, Daniele P. Radicioni, and Valentina Rho. 2017. Dual peccs: a cognitive system for conceptual representation and categorization. *Journal of Experimental & Theoretical Artificial Intelligence*, 29(2):433–452.

Max M Louwerse. 2011. Symbol interdependency in symbolic and embodied cognition. *Topics in Cognitive Science*, 3(2):273–302.

Enrico Mensa, Aureliano Porporato, and Daniele P. Radicioni. 2018a. Annotating Concept Abstractness by Common-sense Knowledge. In *Proceedings of the 17th International Conference of the Italian Association for Artificial Intelligence*, LNAI, Cham. Springer.

Enrico Mensa, Aureliano Porporato, and Daniele P. Radicioni. 2018b. Grasping metaphors: Lexical semantics in metaphor analysis. In *The Semantic Web: ESWC 2018 Satellite Events*, pages 192–195, Cham. Springer International.

Enrico Mensa, Daniele P. Radicioni, and Antonio Lieto. 2018c. COVER: a linguistic resource combining common senseand lexicographic information. *Lang Resources & Evaluation*.

George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. 1990. Introduction to WordNet: An on-line lexical database. *International journal of lexicography*, 3(4):235–244.

George A Miller, Claudia Leacock, Randee Tengi, and Ross T Bunker. 1993. A semantic concordance. In *Proceedings of the workshop on Human Language Technology*, pages 303–308. Association for Computational Linguistics.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Yair Neuman, Dan Assaf, Yohai Cohen, Mark Last, Shlomo Argamon, Newton Howard, and Ophir Frieder. 2013. Metaphor identification in large texts corpora. *PloS one*, 8(4):e62343.

Allen Newell. 1980. Physical symbol systems. *Cognitive science*, 4(2):135–183.

Gabriel Recchia and Michael Jones. 2012. The semantic richness of abstract concepts. *Frontiers in Human Neuroscience*, 6:315.

DL Theijssen, H van Halteren, LWJ Boves, and NHJ Oostdijk. 2011. On the difficulty of making concreteness concrete. *CLIN Journal*, pages 61–77.

Peter D Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. 2011. Literal and metaphorical sense identification through concrete and abstract context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 680–690.

Peter D Turney. 2006. Similarity of semantic relations. *Computational Linguistics*, 32(3):379–416.

Xing Xing, Yi Zhang, and Mei Han. 2010. Query difficulty prediction for contextual image retrieval. In *European Conference on Information Retrieval*, pages 581–585.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd international conference on computational linguistics*, pages 1353–1361. Association for Computational Linguistics.

Effective Communication without Verbs? Sure! Identification of Nominal Utterances in Italian Social Media Texts

Gloria Comandini
Università di Trento
Trento, Italy
gloria.comandini@unitn.it

Manuela Speranza, Bernardo Magnini
Fondazione Bruno Kessler
Trento, Italy
{manspera,magnini}@fbk.eu

Abstract

English. Nominal utterances are very frequent, especially in social media texts, and play a crucial role as they are very dense from a semantic point of view. In spite of this, their automatic identification has received little to no attention. We have thus developed a framework for the annotation of nominal utterances and created the manually annotated corpus COSMIANU (Corpus Of Social Media Italian Annotated with Nominal Utterances), which could be used to train automatic systems.

Italiano. *Gli enunciati nominali sono un fenomento linguistico molto frequente, specialmente nello scritto dei social media, e di cruciale importanza, data la loro alta densità semantica. Tuttavia, ben poca attenzione è stata dedicata al loro riconoscimento automatico. In quest'ottica, questo lavoro illustra le guidelines per l'annotazione manuale degli enunciati nominali da noi sviluppate e presenta il corpus dell'italiano dei social media da noi annotato con gli enunciati nominali (COSMIANU), utilizzabile per addestrare sistemi automatici.*

1 Introduction

Syntactic declarative constructions built around a non-verbal head (as in, for example, "What a nice movie!") are very common linguistic phenomena in many Indo-European, Slavic and Semitic languages (such as Latin, Hebrew, Arabic, Russian, English, Spanish, and Italian), as well as in Finno-Ugric and Bantu languages (Benveniste, 1990; Simone, 2013). Not all of these nominal constructions can be unanimously considered sentences, although they can surely be considered utterances, defined as concrete units of actually produced text, devoid of any pre-determined syntactic or semantic form (Sabatini and Coletti, 1997; Adger, 2003; Graffi, 2012; Ferrari, 2014).

It has been clearly shown that nominal utterances (NUs) occur with relatively high frequency not only in spoken language (Cresti, 1998; Landolfi et al., 2010; Garcia-Marchena, 2016) but also in written texts. Literary and journalistic prose certainly offer some fine examples of NUs (Mortara Garavelli, 1971; Dardano and Trifone, 2001), but nonetheless texts produced with computer mediated communication (CMC) or, more generally, within social media, are also a fertile ground for this phenomenon. In fact, NUs are extremely important from the semantic point of view as they allow speakers or writers to provide a lot of information using only a few words (high semantic density), often without any explicit hierarchical relationship (Sornicola, 1981; Ferrari, 2011a), which is a typical feature of CMC (Ferrari, 2011b).

Yet NUs pose significant challenges when it comes to both their automatic processing, because of the absence of a verbal head, and identification, due to the fact that they can have diverse syntactic structures, containing, for example, dependent clauses with finite verbs.

So far, little or no attention has been paid to the identification and processing of NUs in NLP areas such as information extraction/retrieval, sentiment analysis, and opinion mining. However, in order to address newly emerging challenges, these research fields could greatly benefit from tackling NUs specifically. This is the case, for instance, with aspect-based sentiment analysis, which aims to identify the main (e.g., the most frequently discussed) aspects (e.g., food, service) of given target entities (e.g., restaurants) and the sentiment expressed towards each aspect, instead of detecting the overall polarity of a text span (as sentiment analysis usually does). Similarly, argumen-

tation mining, which takes one step forward with respect to opinion mining by extracting not only information about people's attitudes and opinions, but also about the arguments they give in favor of and against their target entities (e.g., products, institutions, politicians, celebrities, etc.), could dramatically improve by focusing on NUs, which are often used, just like slogans, as the most emphatic part of the argumentation.

As a first step towards enabling automatic systems to process NUs, we have developed a complete framework for their annotation, and have created the Corpus Of Social Media Italian Annotated with Nominal Utterances (COSMIANU), which will be freely distributed with a Creative Commons (CC-BY) licence and can therefore be used to train automatic systems.

In this paper, we first summarize the main criteria adopted for the annotation of NUs (Section 3); in Section 4 we describe the annotated corpus; in Section 5 we present the results of some preliminary experiments on automatic identification of NUs, and finally, in Section 6, we draw some conclusions.

2 Related work

The first corpus-based study of NUs was part of the C-ORAL-ROM project, a multilingual (Italian, French, Portuguese and Spanish) corpus composed by 1,200,000 words of spontaneous speech, created in order to describe the prosodic and syntactic structures of romance languages (Cresti et al., 2004).

Relatively similar is the study conducted on the AN.ANA.S Multilingual Treebank, consisting of 21,300 words of spontaneous speech and task-oriented dialogues in Italian, English and Spanish, manually annotated in order to identify verbless clauses (Landolfi et al., 2010).

In more recent work, Garcia-Marchena (2016) uses the Spanish open-source corpus CORLEC[1] to manually identify and classify over 7,000 verbless utterances in a detailed taxonomy.

While the above-mentioned studies all address verbless sentences and clauses, the phenomenon in which we are interested is wider and includes more complex syntactic structures, partly because we address nominal utterances, which is a wider set than verbless utterances (in our perspective, in fact, the main clause of a NU can govern dependent clauses with finite verbs). For this reason we devised a complete annotation framework. Moreover, to the best of our knowledge, our work is the first attempt towards a corpus-based study of NUs on written texts (Cresti (2004), Landolfi et al. (2010), and Garcia-Marchena (2016) address spoken language).

3 Annotation Framework

In the following, we provide a brief summary of the annotation framework we devised for the manual annotation of NUs, which is based on the literature on NUs in Italian (Mortara Garavelli, 1971; Ferrari, 2011a; Ferrari, 2011b). For a thorough description (and plenty of annotated examples), see the document "Linee guida per l'annotazione degli enunciati nominali" (in Italian) [2].

3.1 NU Identification

According to the annotation schema we propose, every utterance whose main clause is non-verbal, i.e. it does not contain a finite verb (see (1)), is marked as a Nominal Utterance (NU); note, however, that a non-verbal main clause can contain non-finite verbs, such as infinitive and/or participial forms and gerunds (see (2), (3), and (4)).

(1) <NU>Felicissima per il suo ritorno!</NU>
[Very happy about his return!]

(2) <NU>Ma impegnarsi di più?</NU>
[Why not put more effort into it?]

(3) <NU>Spariti i negozi, l'edicola, il posteggio.</NU>
[Shops, news stand, and car park, all gone.]

(4) <NU>Facendo due conti.</NU>
[Doing the math.]

3.2 Coordination of main clauses

When the main clause of an utterance bears a coordination relation to another clause, the NU is annotated as follows:

- If both are non-verbal, the extent of the NU includes them both (see (5));

[1] CORLEC, Corpus Oral de Referencia de la Lengua Española Contemporánea, available from: http://www.lllf.uam.es/ING/Corlec.html

[2] This document is available for consultation from http://tiny.cc/auhvvy

- If one is verbal and the other one is non-verbal, the extent of the NU includes only the non-verbal one (see (6)).

(5) <NU>Acqua a dirotto e tutti a casa!</NU>
[Too much rain and everyone home!]

(6) <NU>I lavori prima,</NU> e poi si cena.
[Chores first, and then we'll eat dinner.]

Due to their peculiar syntactic structure, NUs with coordination are further marked with the attribute "verbal-coordinate" (coordination of verbal and non-verbal clauses) or "non-verbal-coordinate" (coordination of non-verbal clauses).

3.3 NUs with subordinate clauses

Non-verbal subordinate clauses are included in the extent of an NU, as in (7), whereas verbal subordinate clauses are not, as in (8) and (9).

(7) <NU>Che bello partire tutti quanti!</NU>
[Great to leave all together!]

(8) <NU>Felice</NU> che ti sia piaciuta.
[Glad you liked it.]

(9) Siccome piove, <NU>tutti a casa.</NU>
[As it is raining, everyone home.]

NUs with verbal subordinate clauses are marked with a specific attribute, i.e., "verbal-subordinate".

3.4 Ellipses

As explained above, NUs are utterances whose main clause is non-verbal, i.e. it does not contain a finite verb. Unlike in other NUs, in ellipses it is always possible to infer the omitted verb (Mortara Garavelli, 1971; Ferrari, 2010), since the omitted verb is exactly the same as the one in the preceeding utterance.

Ellipses are marked, using the specific attribute "ellipsis", both when the preceeding utterance is written by a different user, as in (10) and when it is written by the same user, as in (11).

(10) Cosa vorresti per cena? [What would you like for dinner?]
<NU>Una pizza!</NU> [A pizza!]

(11) Cosa voglio??? [What do I want???]
<NU>Del rispetto!</NU> [Some respect!]

	#sentences	#words	#tokens
Blogs	1,178	16,054	18,874
Forums	1,331	15,168	18,105
Newsgroups	1,395	15,045	19,109
Soc. networks	1,057	7,770	9,923
Total	4,961	54,039	66,011

Table 1: Data about COSMIANU.

4 Annotations in COSMIANU

COSMIANU contains texts taken from the Web2Corpus_IT (Chiari and Canzonetti, 2014), a balanced Italian corpus of 1,050,000 words consisting of social media texts of five types, i.e., blogs, forums, newsgroups, chats, and social networks. In particular, we focused on semi-synchronous forms of CMC, i.e. blogs, forums, newsgroups, and social networks (Pistolesi, 2004), and randomly chose 24 files (six from each of the four selected categories), for a total of 54,039 words.

These texts consist of discussions between users across a large number of themes (from politics to popular singers). Thus in most cases, users interact with each other creating a dialogic enviroment rich in verbal crossfires and quotes. This kind of interactions are a particularly fertile ground for ellipses and NUs in the form of greetings, which are usually very frequent in spoken language.

Automatic pre-proccessing of the corpus, for which we used the TextPro suite of NLP tools (Pianta et al., 2008), consisted of tokenization and sentence-splitting and resulted in 4,961 sentences and 66,011 tokens (see Table 1 for more detailed data).

The manual annotation was then performed by an expert annotator using the Content Annotation Tool (CAT) (Bartalesi Lenzi et al., 2012). The annotation effort, for an expert annotator, consisted of two weeks of work.

In order to evaluate the inter-annotator agreement, a subpart of the corpus consisting of 5,193 tokens was annotated by a second annotator. The resulting Dice coefficient is 87.40. Both annotators identified 127 NUs, 111 of which are common (evaluation based on exact match).

Table 2 reports, for both the whole corpus and for each subcategory, the total number of NUs and the number of NUs marked with each specific attribute, i.e. "verbal-coordinate", "non-verbal-

	NUs	Verbal coord.	Non-verb. coord.	Verbal subord.	Ellipsis	Simple NUs
Blogs	261	30	15	32	37	194
Forums	263	36	13	23	34	190
Newsgroups	196	33	21	17	35	122
Social networks	304	41	9	19	31	231
Total	1,024	140	58	91	137	737

Table 2: Distribution of NUs in the four social media categories.

	Verbal coord.	Non-verb. coord.	Verbal subord.	Ellipsis
Verbal coord.	-	7	13	38
Non-verb. coord.	7	-	11	10
Verbal subord.	13	11	-	26
Ellipsis	38	10	26	-
no other attribute	82	30	41	63
Total	140	58	91	137

Table 3: Attribute co-occurrence.

coordinate", "verbal-subordinate", and "ellipsis" (NUs that are not marked with any attribute, such as (1), (2), (3), and (4), are referred to as "simple NUs").[3]

In the whole corpus we annotated 1,024 NUs, which means that 20,6% of the sentences contain an NU. This percentage is lower than those reported by Cresti (2004) (38,1%) and Landolfi et al. (2010) (28%). This can be explained by the fact that the above-mentioned studies focus on spoken language, where interrupted strings, brachyologies and turn-taking cues are more frequent with respect to written language. Still, this percentage shows that the nominal style is well represented in written informal Italian, most likely due to its linguistic economy and to its high semantic density, which are particularly useful for expressing emphasis (see (12)).

(12) <NU>Dichiarazione da Mr. Hyde!</NU>
[A statement worthy of Mr. Hyde!]

In addition, the large number of NUs marked as coordinate, either "verbal" (140 NUs) or "non-verbal" (58 NUs) shows that parataxis is constant throughout these texts. In fact, NUs appear to be extremely suitable to the parataxis typical of CMC; furthermore, they are often isolated, i.e., free from hierarchical syntactic bonds. This also explains why NUs can be composed of a series of denotative elements simply listed without any explicit hierarchical bond, as in (13), in a way that reminds one of a list of keywords.

(13) <NU>Buon senso, etica, vincere tanto per vincere.</NU>
[Common sense, ethics, winning for winning's sake.]

Looking at the distribution of NUs in the four subcategories, we see that social networks have the highest number of NUs (304), despite having a significantly lower number of tokens than blogs, forums and newsgroups. This probably depends on the high perceived communicative economy typical of social networks (Cosenza, 2014), which leads writers to produce short, almost telegraphic, texts.

In Table 3 we report the co-occurence of NU attributes by pairs[4] in order to show how diverse syntactic structures NUs can have. Particularly interesting is the presence of 38 NUs containing ellipses coordinated with a verbal clause; in fact, the ellipsis usually follows the verbal clause, whose verb is implied in a contrastive context. Additionally, ellipses can support a verbal subordinate clause (in our corpus we have 26 cases), which usually adds further information in favor of the contrastive utterance (see (14)).

[3]Notice that a single NU can be marked with more than one attribute.

[4]Although we have case where NUs have been marked with up to four attributes, we only focus on co-occurrence by attribute pairs.

(14) Non è un edificio specifico, <NU> ma una tipologia architettonica </NU> che caratterizza l'URSS.
[It is not a specific building, but an architectural typology that characterizes the USSR.]

5 Automatic Identification of NUs

We used COSMIANU to train an open source SVM classifier, YamCha[5], and performed some preliminary experiments on NU identification. As training data, we selected 44,170 tokens (i.e. about 2/3 of the corpus) while maintaining the same proportion of blogs, forums, newsgroups, and social networks over the whole corpus. We used the remaining part of the corpus (21,841 tokens) as a test set. In these preliminary experiments we also included the NUs that appear in the text as metadata, which are annotated and marked with the specific tag "metadata" in COSMIANU, as shown in Example (15) [6]. The training set and the test set thus contain respectively 1,775 and 1,058 NUs.

(15) <NU> Data: 27/09/2010. </NU>
[Date: 09/27/2010.]

We pre-processed the data using the TextPro suite (Pianta et al., 2008) and performed a number of experiments combining the following basic features: two-word window context (W2), three-word window context (W3), token (Tok), lemma (Lem), and Part-of-Speech (Pos).

Configuration	Prec.	Rec.	F1
Baseline	33.80	27.13	30.10
W2+Tok+Lem+Pos	79.80	67.96	73.40

Table 4: Results on NU identification.

Table 4 reports, in terms of Precision, Recall, and F1, the results we obtained with the baseline configuration (the system identifies only the NUs in the test set that also appear in the training set) and those we obtained with the best configuration, i.e. using all the features and a two-word window context. With the latter, the classifier identified 901 NUs, of which 719 are correct (exact match), thus reaching an F1 of 73.40% and outperforming the baseline by over 43 points.

[5]Yet Another Multipurpose CHunk Annotator. Website: http://chasen.org/ taku/software/yamcha/

[6]Metadata usually refer to when and where a certain message has been written; although "metadata" NUs are very frequent in the corpus (more than 60% of the total), they are not particularly interesting from a linguistic point of view and we did not include them in the counts of Section 4.

6 Conclusion and Future Work

This work shows how common NUs are in written informal language, as well as how important they are in conveying semantically dense concepts in emphatic informative peaks, which could be useful for many NLP fields (e.g., argumentation mining and aspect-based sentiment analysis).

By creating COSMIANU, an Italian corpus annotated with NUs, and making it freely available to the research community, we made a first step towards the development of automatic tools for the identification and classification of NUs. In our preliminary experiments on NU identification (performed using an SWM classifier), with our best configuration, we obtained a performance of 73.40% in terms of F1 on all NUs (i.e. including metadata).

In the future, we intend to further expand COSMIANU, both in terms of its size and in terms of the annotations it includes, hoping that this will encourage more research on this extremely common, and yet almost neglected, linguistic phenomenon. We also plan to work on the analysis and automatic recognition of NUs, especially when they are used to convey hate speech, in the form of racist, sexist, homo/transphobic or classist slogans and insults.

Acknowledgments

We would like to thank Isabella Chiari for providing us the Web2Corpus_IT, from which we selected the raw texts to build COSMIANU. We also thank our colleagues Roberto Zanoli and Rachele Sprugnoli for their valuable advice and contributions in performing the experiments and defining the annotation guidelines.

References

David Adger. 2003. *Core Syntax: A Minimalist Approach*. Oxford University Press.

Valentina Bartalesi Lenzi, Giovanni Moretti, and Rachele Sprugnoli. 2012. CAT: the CELCT Annotation Tool. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'12)*, pages 333–338, Istanbul, Turkey, May. European Language Resources Association (ELRA).

Émile Benveniste. 1990. *Problemi di linguistica generale*. Mondadori, Milano, Italia.

Isabella Chiari and Alessio Canzonetti. 2014. Le forme della comunicazione mediata dal computer: generi, tipi e standard di annotazione. In E. Garavelli and E. Suomela-Härmä, editors, *Dal manoscritto al web: canali e modalità di trasmissione dell'italiano*, pages 595–606. Franco Cesati Editore, Firenze, Italia.

Giovanna Cosenza. 2014. *Introduzione alla semiotica dei nuovi media*. Laterza, Bari, Italia.

Emanuela Cresti, Fernanda Bacelar do Nascimento, Antonio Moreno-Sandoval, Jean Véronis, Philippe Martin, and Khalid Choukri. 2004. The C-ORAL-ROM CORPUS. A Multilingual Resource of Spontaneous Speech for Romance Languages. In Maria Teresa Lino, Maria Francisca Xavier, Fátima Ferreira, Rute Costa, and Raquel Silva, editors, *Prococeedings of the 4th LREC Conference*, pages 575–578, Paris, France. European Language Resources Association (ELRA).

Emanuela Cresti. 1998. Gli enunciati nominali. In M. T. Navarro, editor, *Atti del IV convegno internazionale SILFI (Madrid 27-29 giugno 1996)*, pages 171–191, Pisa. Franco Cesati Editore.

Maurizio Dardano and Pietro Trifone. 2001. *La nuova grammatica della lingua italiana*. Zanichelli, Milano, Italia.

Angela Ferrari. 2010. Enunciati ellittici. *Enciclopedia dell'Italiano*. `http://www.treccani.it/enciclopedia/enunciati-ellittici_(Enciclopedia-dell'Italiano)/`.

Angela Ferrari. 2011a. Enunciati nominali. *Enciclopedia dell'Italiano*. `http://www.treccani.it/enciclopedia/enunciati-nominali_(Enciclopedia_dell'Italiano)/`.

Angela Ferrari. 2011b. Stile nominale. *Enciclopedia dell'Italiano*. `http://www.treccani.it/enciclopedia/stile-nominale_(Enciclopedia-dell'Italiano)/`.

Angela Ferrari. 2014. *Linguistica del testo. Principi, fenomeni, strutture*. Carocci, Roma, Italia.

Oscar Garcia-Marchena. 2016. Spanish Verbless Clauses and Fragments. A corpus analysis. In Antonio Moreno Ortiz and Chantal Pérez-Hernández, editors, *CILC 2016. 8th International Conference on Corpus Linguistics*, volume 1 of *EPiC Series in Language and Linguistics*, pages 130–143. EasyChair.

Giorgio Graffi. 2012. *La frase: l'analisi logica*. Carocci, Roma, Italia.

Annamaria Landolfi, Carmela Sammarco, and Miriam Voghera. 2010. Verbless clauses in Italian, Spanish and English: a Treebank annotation. In S. Bolasco, I. Chiari, and L. Giuliano, editors, *Statistical Analysis of Textual Data. Proceedings of the 10th International Conference on Statistical Analysis of Textual Data (JADT 2010)*, pages 450–459. Roma, Italia, June 9-11.

Bice Mortara Garavelli. 1971. Fra norma e invenzione: lo stile nominale. In Accademia della Crusca, editor, *Studi di grammatica italiana*, volume 1, pages 271–315. G. C. Sansoni Editore, Firenze, Italia.

Emanuele Pianta, Christian Girardi, and Roberto Zanoli. 2008. The TextPro tool suite. In *Proceedings of LREC, 6th edition of the Language Resources and Evaluation Conference*, Marrakech, Morocco, May 28-30.

Elena Pistolesi. 2004. *Il parlar spedito. L'italiano di chat, e-mail e sms*. Esedra, Padova, Italia.

Francesco Sabatini and Vittorio Coletti. 1997. *Dizionario Italiano Sabatini-Coletti*. Giunti, Firenze, Italia.

Raffaele Simone. 2013. *Nuovi fondamenti di linguistica*. McGraw-Hill, Milano, Italia.

Rosanna Sornicola. 1981. *Sul parlato*. Il Mulino, Bologna, Italia.

On the Readability of Deep Learning Models: the role of Kernel-based Deep Architectures

Danilo Croce and **Daniele Rossini** and **Roberto Basili**
Department Of Enterprise Engineering
University of Roma, Tor Vergata
{croce,basili}@info.uniroma2.it

Abstract

English. Deep Neural Networks achieve state-of-the-art performances in several semantic NLP tasks but lack of explanation capabilities as for the limited interpretability of the underlying acquired models. In other words, tracing back causal connections between the linguistic properties of an input instance and the produced classification is not possible. In this paper, we propose to apply *Layerwise Relevance Propagation* over linguistically motivated neural architectures, namely *Kernel-based Deep Architectures* (KDA), to guide argumentations and explanation inferences. In this way, decisions provided by a KDA can be linked to the semantics of input examples, used to linguistically motivate the network output.

Italiano. *Le Deep Neural Network raggiungono oggi lo stato dell'arte in molti processi di NLP, ma la scarsa interpretabilitá dei modelli risultanti dall'addestramento limita la comprensione delle loro inferenze. Non é possibile cioé determinare connessioni causali tra le proprietá linguistiche di un esempio e la classificazione prodotta dalla rete. In questo lavoro, l'applicazione della* Layerwise Relevance Propagation *alle* Kernel-based Deep Architecture*(KDA) é usata per determinare connessioni tra la semantica dell'input e la classe di output che corrispondono a spiegazioni linguistiche e trasparenti della decisione.*

1 Introduction

Deep Neural Networks are usually criticized as they are not epistemologically transparent devices, i.e. their models cannot be used to provide explanations of the resulting inferences. An example can be neural question classification (QC) (e.g. (Croce et al., 2017)). In QC the correct category of a question is detected to optimize the later stages of a question answering system, (Li and Roth, 2006). An epistemologically transparent learning system should trace back the causal connections between the proposed question category and the linguistic properties of the input question. For example, the system could motivate the decision: *"What is the capital of Zimbabwe?" refers to a* `Location`, with a sentence such as: *Since it is similar to "What is the capital of California?"* which also refers to a `Location`. Unfortunately, neural models, as for example Multilayer Perceptrons (MLP), Long Short-Term Memory Networks (LSTM), (Hochreiter and Schmidhuber, 1997), or even Attention-based Networks (Larochelle and Hinton, 2010), correspond to parameters that have no clear conceptual counterpart: it is thus difficult to trace back the network components (e.g. neurons or layers in the resulting topology) responsible for the answer.

In image classification, *Layerwise Relevance Propagation* (LRP) (Bach et al., 2015) has been used to decompose backward across the MLP layers the evidence about the contribution of individual input fragments (i.e. pixels of the input images) to the final decision. Evaluation against the MNIST and ILSVRC benchmarks suggests that LRP activates associations between input and output fragments, thus tracing back meaningful causal connections.

In this paper, we propose the use of a similar mechanism over a linguistically motivated network architecture, the Kernel-based Deep Architecture (KDA), (Croce et al., 2017). Tree Kernels (Collins and Duffy, 2001) are here used to integrate syntactic/semantic information within a MLP network. We will show how KDA input nodes correspond to linguistic instances and by applying the LRP method we are able to trace back causal associations between the semantic classification and such instances. Evaluation of the LRP algorithm is based on the idea that explanations

improve the user expectations about the correctness of an answer and shows its applicability in human computer interfaces.

In the rest of the paper, Section 2 describes the KDA neural approach while section 3 illustrates how LRP connects to KDAs. In section 4 early results of the evaluation are reported.

2 Training Neural Networks in Kernel Spaces

Given a training set $o \in D$, a kernel $K(o_i, o_j)$ is a similarity function over D^2 that corresponds to a dot product in the implicit kernel space, i.e., $K(o_i, o_j) = \Phi(o_i) \cdot \Phi(o_j)$. Kernel functions are used by learning algorithms, such as Support Vector Machines (Shawe-Taylor and Cristianini, 2004), to efficiently operate on instances in the kernel space: their advantage is that the projection function $\Phi(o) = \vec{x} \in \mathbb{R}^n$ is never explicitly computed. The Nyström method is a factorization method applied to derive a new low-dimensional embedding $\tilde{x}$ in a l-dimensional space, with $l \ll n$ so that $G \approx \tilde{G} = \tilde{X}\tilde{X}^\top$, where $G = XX^\top$ is the Gram matrix such that $G_{ij} = \Phi(o_i)\Phi(o_j) = K(o_i, o_j)$. The approximation $\tilde{G}$ is obtained using a subset of l columns of the matrix, i.e., a selection of a subset $L \subset D$ of the available examples, called *landmarks*. Given l randomly sampled columns of G, let $C \in \mathbb{R}^{|D| \times l}$ be the matrix of these sampled columns. Then, we can rearrange the columns and rows of G and define $X = [X_1 \ X_2]$ such that:

$$G = \begin{bmatrix} W & X_1^\top X_2 \\ X_2^\top X_1 & X_2^\top X_2 \end{bmatrix} = \begin{bmatrix} C \\ X_2^\top X_1 \end{bmatrix}$$

where $W = X_1^\top X_1$, i.e., the subset of G that contains only landmarks. The Nyström approximation can be defined as:

$$G \approx \tilde{G} = CW^\dagger C^\top \tag{1}$$

where $W^\dagger$ denotes the Moore-Penrose inverse of W. If we apply the Singular Value Decomposition (SVD) to W, which is symmetric definite positive, we get $W = USV^\top = USU^\top$. Then it is straightforward to see that $W^\dagger = US^{-1}U^\top = US^{-\frac{1}{2}}S^{-\frac{1}{2}}U^\top$ and that by substitution $G \approx \tilde{G} = (CUS^{-\frac{1}{2}})(CUS^{-\frac{1}{2}})^\top = \tilde{X}\tilde{X}^\top$. Given an example $o \in D$, its new low-dimensional representation $\tilde{\vec{x}}$ is determined by considering the corresponding item of C as

$$\tilde{\vec{x}} = \vec{c}US^{-\frac{1}{2}} \tag{2}$$

where $\vec{c}$ is the vector whose dimensions contain the evaluations of the kernel function between o and each landmark $o_j \in L$. Therefore, the method produces l-dimensional vectors.

Given a labeled dataset, a Multi-Layer Perceptron (MLP) architecture can be defined, with a specific Nyström layer based on the Nyström embeddings of Eq. 2, (Croce et al., 2017).

Such Kernel-based Deep Architecture (KDA) has an *input layer*, a *Nyström layer*, a possibly empty sequence of non-linear *hidden layers* and a final *classification layer*, which produces the output. In particular, the input layer corresponds to the input vector $\vec{c}$, i.e., the row of the C matrix associated to an example o. It is then mapped to the *Nyström* layer, through the projection in Equation 2. Notice that the embedding provides also the proper weights, defined by $US^{-\frac{1}{2}}$, so that the mapping can be expressed through the Nyström matrix $H_{Ny} = US^{-\frac{1}{2}}$: it corresponds to a pre-training stage based on the SVD. Formally, the low-dimensional embedding of an input example o, $\tilde{\vec{x}} = \vec{c}\, H_{Ny} = \vec{c}\, US^{-\frac{1}{2}}$ encodes the kernel space. Any neural network can then be adopted: in the rest of this paper, we assume that a traditional Multi-Layer Perceptron (MLP) architecture is stacked in order to solve the targeted classification problems. The final layer of KDA is the *classification layer* whose dimensionality depends on the classification task: it computes a linear classification function with a softmax operator.

A KDA is stimulated by an input vector c which corresponds to the kernel evaluations $K(o, l_i)$ between each example o and the landmarks l_i. Linguistic kernels (such as Semantic Tree Kernels (Croce et al., 2011)) depend on the syntactic/semantic similarity between the x and the subset of l_i used for the space reconstruction. We will see hereafter how tracing back through relevance propagation into a KDA architecture corresponds to determine which semantic landmarks contribute mostly to the final output decision.

3 Layer-wise Relevance Propagation in Kernel-based Deep Architectures

Layer-wise Relevance propagation (LRP, presented in (Bach et al., 2015)) is a framework which allows to decompose the prediction of a deep neural network computed over a sample, e.g. an im-

age, down to relevance scores for the single input dimensions, such as a subset of pixels.

Formally, let $f : \mathbb{R}^d \to \mathbb{R}^+$ be a positive real-valued function taking a vector $\vec{x} \in \mathbb{R}^d$ as input: f quantifies, for example, the probability of $\vec{x}$ characterizing a certain class. The Layer-wise Relevance Propagation assigns to each dimension, or feature, x_d, a relevance score $R_d^{(1)}$ such that:

$$f(x) \approx \sum_d R_d^{(1)} \tag{3}$$

Features whose score $R_d^{(1)} > 0$ (or $d\ R_d^{(1)} < 0$) correspond to evidence in favor (or against) the output classification. In other words, LRP allows to identify fragments of the input playing key roles in the decision, by propagating relevance backwards. Let us suppose to know the relevance score $R_j^{(l+1)}$ of a neuron j at network layer $l+1$, then it can be decomposed into messages $R_{i\leftarrow j}^{(l,l+1)}$ sent to neurons i in layer l:

$$R_j^{(l+1)} = \sum_{i\in(l)} R_{i\leftarrow j}^{(l,l+1)} \tag{4}$$

Hence the relevance of a neuron i at layer l can be defined as:

$$R_i^{(l)} = \sum_{j\in(l+1)} R_{i\leftarrow j}^{(l,l+1)} \tag{5}$$

Note that 4 and 5 are such that 3 holds. In this work, we adopted the ϵ-rule defined in (Bach et al., 2015) to compute the messages $R_{i\leftarrow j}^{(l,l+1)}$, i.e.

$$R_{i\leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j + \epsilon \cdot \text{sign}(z_j)} R_j^{(l+1)}$$

where $z_{ij} = x_i w_{ij}$ and $\epsilon > 0$ is a numerical stabilizing term and must be small. Notice that weights w_{ij} correspond to weighted activations of input neurons. If we apply LRP to a KDA it implicitly traces the relevance back to the input layer, i.e. to the landmarks. It thus tracks back syntactic, semantic and lexical relations between a question and the landmark and it grants high relevance to the relations the network selected as highly discriminating for the class representations it learned; note that this is different from similarity in terms of kernel-function evaluation as the latter is task independent whereas LRP scores are not. Notice also that each landmark is uniquely associated to an entry of the input vector $\vec{c}$, as shown in Sec 2, and, as a member of the training dataset, it also corresponds to a known class.

4 Explanatory Models

LRP allows the automatic compilation of justifications for the KDA classifications: explanations are possible using landmarks $\{\ell\}$ as examples. The $\{\ell\}$ that the LRP method produces as the most active elements in layer 0 are semantic analogues of input annotated examples. An *Explanatory Model* is the function in charge of compiling the linguistically fluent explanation of individual analogies (or differences) with the input case. The meaningfulness of such analogies makes a resulting explanation clear and should increase the user confidence on the system reliability. When a sentence o is classified, LRP assigns activation scores r_ℓ^s to each individual landmark ℓ: let $L^{(+)}$ (or $L^{(-)}$) denote the set of landmarks with positive (or negative) activation scores.

Formally, an explanation is characterized by a triple $e = \langle s, C, \tau \rangle$ where s is the input sentence, C is the predicted label and τ is the modality of the explanation: $\tau = +1$ for positive (i.e. acceptance) statements while $\tau = -1$ correspond to rejections of the decision C. A landmark ℓ is *positively activated* for a given sentence s if there are not more than $k-1$ other active landmarks[1] ℓ' whose activation value is higher than the one for ℓ, i.e.

$$|\{\ell' \in L^{(+)} : \ell' \neq \ell \wedge r_{\ell'}^s \geq r_\ell^s > 0\}| < k$$

A landmark is *negatively activated* when: $|\{\ell' \in L^{(-)} : \ell' \neq \ell \wedge r_{\ell'}^s \leq r_\ell^s < 0\}| < k$. Positively (or negative) active landmarks in L_k are assigned to an activation value $a(\ell, s) = +1$ (-1). For all other not activated landmarks: $a(\ell, s) = 0$.

Given the explanation $e = \langle s, C, \tau \rangle$, a landmark ℓ whose (known) class is C_ℓ is *consistent* (or *inconsistent*) with e according to the fact that the following function:

$$\delta(C_\ell, C) \cdot a(\ell, q) \cdot \tau$$

is positive (or negative, respectively), where $\delta(C', C) = 2\delta_{kron}(C' = C) - 1$ and δ_{kron} is the Kronecker delta.

The *explanatory model* is then a function $\mathcal{M}(e, L_k)$ which maps an explanation e, a sub set L_k of the active *and* consistent landmarks L for e into a sentence in natural language. Of course several definitions for $\mathcal{M}(e, L_k)$ and L_k are possible.

[1] k is a parameter used to make explanation depending on not more than k landmarks, denoted by L_k.

A general explanatory model would be:

$$\mathcal{M}(e, L_k) = \begin{cases} \text{“ } s \text{ is } C \text{ since it is similar to } \ell \text{ ”} \\ \forall \ell \in L_k^+ \quad \text{if } \tau > 0 \\ \text{“ } s \text{ is not } C \text{ since it is different} \\ \text{from } \ell \text{ which is } C \text{ ”} \\ \forall \ell \in L_k^- \quad \text{if } \tau < 0 \\ \text{“ } s \text{ is } C \text{ but I don't know why ”} \\ \text{if } L_k = \emptyset \end{cases}$$

where L_k^+,$L_k^- \subseteq L_k$ are the partitions of landmarks with positive (and negative) relevance scores in L_k, respectively. Here we provide examples for two explanatory models, used during the experimental evaluation. A first possible model returns the analogy only with the (unique) consistent landmark with the highest positive score if $\tau = 1$ and lowest negative when $\tau = -1$. The explanation of a rejected decision in the Argument Classification of a Semantic Role Labeling task (Vanzo et al., 2016), described by the triple $e_1 = \langle$'*vai in camera da letto*', SOURCE$_{\text{BRINGING}}$, $-1\rangle$, is:

*I think "**in camera da letto**" IS NOT* [SOURCE] of [BRINGING] *in "Vai in camera da letto"* (LU:[*vai*]) *since it's different from "sul tavolino" which is* [SOURCE] of [BRINGING] *in "Portami il mio catalogo sul tavolino"* (LU:[*porta*])

The second model uses two active landmarks: one consistent and one contradictory with respect to the decision. For the triple $e_1 = \langle$'*vai in camera da letto*', GOAL$_{\text{MOTION}}$, $1\rangle$ the second model produces:

I think "**in camera da letto**" IS [GOAL] of [MOTION] *in "Vai in camera da letto"* (LU:[*vai*]) *since it recalls **"al telefono"** which is* [GOAL] of [MOTION] *in "Vai al telefono e controlla se ci sono messaggi"* (LU:[*vai*]) *and it IS NOT* [SOURCE] of [BRINGING] *since different from "sul tavolino" which is the* [SOURCE] of [BRINGING] *in "Portami il mio catalogo sul tavolino"* (LU:[*portami*])

4.1 Evaluation methodology

In order to evaluate the impact of the produced explanations, we defined the following task: given a classification decision, i.e. the input *o* is classified as C, to measure the impact of the explanation e on the belief that a user exhibits on the statement "$o \in C$ is `true`". This information can be modeled through the estimates of the following probabilities: $P(o \in C)$ that characterizes the amount of confidence the user has in accepting the statement, and its corresponding form $P(o \in C|e)$, i.e. the same quantity in the case the user is provided by the explanation e. The core idea is that semantically coherent and exhaustive explanations must indicate correct classifications whereas incoherent or non-existent explanations must hint towards wrong classifications. A quantitative measure of such an increase (or decrease) in confidence is the Information Gain (IG, (Kononenko and Bratko, 1991)) of the decision $o \in C$. Notice that IG measures the increase of probability corresponding to correct decisions, and the reduction of the probability in case the decision is wrong. This amount suitably addresses the shift in uncertainty $-log_2(P(\cdot))$ between two (subjective) estimates, i.e., $P(o \in C)$ vs. $P(o \in C|e)$.

Different explanatory models $\mathcal{M}$ can be also compared. The relative Information Gain $I_{\mathcal{M}}$ is measured against a collection of explanations $e \in T_{\mathcal{M}}$ generated by $\mathcal{M}$ and then normalized throughout the collection's entropy $\mathcal{E}$ as follows:

$$I_{\mathcal{M}} = \frac{1}{\mathcal{E}} \frac{1}{\mid T_{\mathcal{M}} \mid} \sum_{e \in T_{\mathcal{M}}} I(e)$$

where $I(e)$ is the IG of each explanation[2].

5 Experimental Evaluation

The effectiveness of the proposed approach has been measured against two different semantic processing tasks, i.e. Question Classification (QC) over the UIUC dataset (Li and Roth, 2006) and Argument Classification in Semantic Role Labeling (SRL-AC) over the HuRIC dataset (Bastianelli et al., 2014; Vanzo et al., 2016). The adopted architecture consisted in a LRP-integrated KDA with 1 hidden layers and 500 landmarks for QC, 2 hidden layers and 100 landmarks for SRL-AC and a stabilization-term $\epsilon = 10e^{-8}$.

We defined five quality categories and associated each with a value of $P(o \in C|e)$, as shown in Table 1. Three annotators then independently rated explanations generated from a collection composed of an equal number of correct and wrong classifications (for a total amount of 300 and 64 explanations, respectively, for QC and SRL-AC). This perfect balancing makes the prior probability $P(o \in C)$ being 0.5, i.e. maximal entropy with a baseline IG $= 0$ in the $[-1, 1]$ range. Notice that annotators had no information on the

[2]More details are in (Kononenko and Bratko, 1991)

Category	$P(o \in C\|e)$	$1-P(o \in C\|e)$
V.Good	0.95	0.05
Good	0.8	0.2
Weak	0.5	0.5
Bad	0.2	0.8
Incoher.	0.05	0.95

Table 1: Posterior probab. w.r.t. quality categories

Model	QC	SRL-AC
One landmark	0.548	0.669
Two landmarks	0.580	0.784

Table 2: Information gains for two Explanatory Models applied to the QC and SRL-AC datasets.

system classification performance, but just knowledge of the explanation dataset entropy.

5.1 Question Classification

Experimental evaluations[3] showed that both the models were able to gain more than half the bit required to ascertain whether the network statement is true or not (Table 2). Consider:

I think "What year did Oklahoma become a state ?" refers to a NUMBER *since recalls me "The film Jaws was made in what year ?"*

Here the model returned a coherent supporting evidence, a somewhat easy case as for the available discriminative pair, i.e. *"What year"*. The system is able to capture semantic similarities even in poorer conditions, e.g.:

I think "Where is the Mall of the America ?" refers to a LOCATION *since recalls me "What town was the setting for The Music Man ?" which refers to a* LOCATION*.*

This high quality explanation is achieved even if with such poor lexical overlap. It seems that richer representations are here involved with grammatical and semantic similarity used as the main information involved in the decision at hand. Let us consider:

I think "Mexican pesos are worth what in U.S. dollars ?" refers to a DESCRIPTION *since it recalls me "What is the Bernoulli Principle ?"*

Here the provided explanation is incoherent, as expected since the classification is wrong. Now consider:

I think "What is the sales tax in Minnesota ?" refers to a NUMBER *since it recalls me "What is the population of Mozambique ?" and does not refer to a* ENTITY *since different from "What is a fear of slime ?".*

[3]For details on KDA performance against the task, see (Croce et al., 2017)

Although explanation seems fairly coherent, it is actually misleading as ENTITY is the annotated class. This shows how the system may lack of contextual information, as humans do, against inherently ambiguous questions.

5.2 Argument Classification

Evaluation also targeted a second task, that is Argument classification in Semantic Role Labeling (SRL-AC): KDA is here fed with vectors from tree kernel evaluations as discussed in (Croce et al., 2011). The evaluation is carried out over the HuRIC dataset (Vanzo et al., 2016), including about 240 domotic commands in Italian, comprising of about 450 roles. The system has an accuracy of 91.2% on about 90 examples, while the training and development set have a size of, respectively, 270 and 90 examples. We considered 64 explanations for measuring the IG of the two explanation models. Table 2 confirms that both explanatory models performed even better than in QC. This is due to the narrower linguistic domain (14 frames are involved) and the clearer boundaries between classes: annotators seem more sensitive to the explanatory information to assess the network decision. Examples of generated sentences are:

I think **"con me"** *is NOT the* MANNER *of* COTHEME *in "Robot vieni* ***con me*** *nel soggiorno*? (LU:[*vieni*])" *since it does NOT recall me* **"lentamente"** *which is* MANNER *in "Per favore segui quella persona* ***lentamente*** (LU:[*segui*])". *It is rather* COTHEME *of* COTHEME *since it recalls me* **"mi"** which is COTHEME *in "Segui***mi** *nel bagno* (LU:[*segui*])".

6 Conclusion and Future Works

This paper describes an LRP application to a KDA that makes use of analogies as explanations of a neural network decision. A methodology to measure the explanation quality has been also proposed and the experimental evidence confirms the effectiveness of the method in increasing the trust of a user upon automatic classifications. Future work will focus on the selection of subtrees as meaningful evidences for the explanation, or on the modeling of negative information for disambiguation as well as on more in depth investigation of the landmark selection policies. Moreover, improved experimental scenarios involving users and dialogues will be also designed, e.g. involving further investigation within Semantic Role Labeling, using the method proposed in (Croce et al., 2012).

References

Sebastian Bach, Alexander Binder, Gregoire Montavon, Frederick Klauschen, Klaus-Robert Mller, and Wojciech Samek. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLOS ONE*, 10(7).

Emanuele Bastianelli, Giuseppe Castellucci, Danilo Croce, Luca Iocchi, Roberto Basili, and Daniele Nardi. 2014. Huric: a human robot interaction corpus. In *LREC*, pages 4519–4526. European Language Resources Association (ELRA).

Michael Collins and Nigel Duffy. 2001. New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL '02), July 7-12, 2002, Philadelphia, PA, USA*, pages 263–270. Association for Computational Linguistics, Morristown, NJ, USA.

Danilo Croce, Alessandro Moschitti, and Roberto Basili. 2011. Structured lexical similarity via convolution kernels on dependency trees. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1034–1046. Association for Computational Linguistics.

Danilo Croce, Alessandro Moschitti, Roberto Basili, and Martha Palmer. 2012. Verb classification using distributional similarity in syntactic and semantic structures. In *ACL (1)*, pages 263–272. The Association for Computer Linguistics.

Danilo Croce, Simone Filice, Giuseppe Castellucci, and Roberto Basili. 2017. Deep learning in semantic kernel spaces. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 345–354, Vancouver, Canada, July. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November.

Igor Kononenko and Ivan Bratko. 1991. Information-based evaluation criterion for classifier's performance. *Machine Learning*, 6(1):67–80, Jan.

Hugo Larochelle and Geoffrey E. Hinton. 2010. Learning to combine foveal glimpses with a third-order boltzmann machine. In *Proceedings of Neural Information Processing Systems (NIPS)*, pages 1243–1251.

Xin Li and Dan Roth. 2006. Learning question classifiers: the role of semantic information. *Natural Language Engineering*, 12(3):229–249.

John Shawe-Taylor and Nello Cristianini. 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge, UK.

Andrea Vanzo, Danilo Croce, Roberto Basili, and Daniele Nardi. 2016. Context-aware spoken language understanding for human robot interaction. In *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.*

The CHROME Manifesto: integrating multimodal data into Cultural Heritage Resources

Francesco Cutugno
Università degli Studi di Napoli "Federico II"
cutugno@unina.it

Felice Dell'Orletta
Istituto di Linguistica Computazionale del CNR - Pisa
ItaliaNLP Lab - *www.italianlp.it*
felice.dellorletta@ilc.cnr.it

Isabella Poggi
Università di Roma3
isabella.poggi@uniroma3.it

Renata Savy
Università degli Studi di Salerno
rsavy@unisa.it

Antonio Sorgente
Istituto di Scienze Applicate e Sistemi Intelligenti del CNR – Pozzuoli
antonio.sorgente@cnr.it

Abstract

English. The CHROME Project aims at collecting a wide portfolio of digital resources oriented to technological application in Cultural Heritage (henceforth CH). The contributions for the realisation of such objective come from the efforts of computer scientists, psychologists, architects, and computational linguists, who constitute an interdisciplinary equipe. We are collecting and analyzing texts, spoken materials, architectural surveys, and human motion videos, attempting the integration of these data in a multidimensional platform based on multilevel annotation systems, game engines importing, and virtualization techniques. As case of study we choose to work on the magic travel along three Charterhouses located in Campania region: S. Martino in Naples, S. Lorenzo in Padula (Salerno) and S. Giacomo, in Capri.

Italiano. *Il progetto CHROME (Cultural Heritage Resources Orienting Multimodal Experiences – PRIN 2015 MIUR) si pone come scopo la raccolta di una ampia gamma di risorse digitali da utilizzare in applicazione tecnologiche per il miglioramento della fruizione dei beni culturali (CH). A questo obiettivo concorrono interdisciplinarmente informatici, psicologi, architetti, linguisti che collezionano testi, registrazioni di parlato, rilievi architettonici, video e human motion capture. Questi dati sono poi integrati in una piattaforma nella quale è possibile effettuare una annotazione multidimensionale, sono anche utilizzati per la virtualizzazione di ambienti tridimensionali e il porting in ambienti di gaming.*

1 Introduction

The CHROME project was born with the intention of creating a framework and methodology to collect, represent and analyze cultural heritage contents and present them through artificial agents whose behavior is inspired by accurate analysis of expert guides, museum curators and tour operators. These gatekeepers are those professional figures possessing a significant amount of knowledge concerning how people should be guided in the exploration of cultural contents. In this sense, they act as mediators between cultural heritage and visitors by using a set of communication strategies, both verbal and non-verbal, aimed at maintaining a high level of engagement and delivering high-quality content.

The overall experience of accessing cultural heritage is greatly enriched by these professional figures: their knowledge and experience, therefore, should not be overlooked when designing artificial agents oriented to cultural heritage presentation. As this knowledge is primarily based on experience collected on the field, the CHROME project aims at recording the performance of gatekeepers in a sensible environment so that formal analysis of their behavior can be documented and studied. The result of this process (see Fig. 1), conducted jointly by humanities and computer scientists, will lead to the formali-

zation of a model describing the behaviors adopted by gatekeepers when presenting cultural heritage. This will then be used to control a humanoid robot designed to follow similar presentation strategies. Taking in account this aim, the main goals of the project are to: **collect** and provide the scientific community with reference datasets to study human-human interaction during the presentation of cultural heritage by professionals; **investigate** the structure of the texts contained in the collected corpus in order to produce automatic approaches supporting text generation for oral presentations in cultural heritage domain; **provide** a reference computational model to support development of artificial agents exhibiting coherent and engaging behavioural strategies. In addition to the orality degree of the assembled presentations, special attention will be attributed to non-verbal aspects. Specifically, CHROME will concentrate on enriching the presentation with consistent prosody and gestures. Finally, another goal is to **evaluate** the impact of these agents in simplifying access to cultural heritage and attract visitors in cultural sites.

For the realization of such goals, five research groups are involved in the CHROME projects covering different scientific and humanistic disciplines that complement each other. The equipe is highly interdisciplinary and is formed of linguists (with specific competences in prosody, pragmatics, paralinguistics, and non-verbal behavior analysis), computational linguists and computer scientists (with skills in Artificial Intelligence and Human Machine Interaction) The teams involved in the project are**:**

- **UrbanEco** (Naples – Federico II) an interdisciplinary team formed by computer scientists, architects, linguists, aiming at collecting 3D architectural surveys and speech and gesture corpora. UrbanEco is also designing multimodal interaction systems; sub-partner linked to this unit is the "**Polo Museale della Campania - MiBaCT**" the local section of the Italian Cultural Ministry managing more than 30 museums in our region;
- **ILC** (Pisa – CNR) will develop systems for automatically extracting and organizing linguistic and domain knowledge from domain-specific corpora;
- **UniSa** (University of Salerno) will analyze texts and will afford the theme of prosodic analysis of spoken material finalized at speech synthesis issue;
- **ISASI** (Pozzuoli, CNR) will afford the challenge of CH question answering and language generation for the realization of interaction models in natural language;
- **RomaTre** (Roma, University RomaTre), will confront the theme of multimodal communication and gesture analysis.

As case of study we choose to work on the magic travel along three Charterhouses located in Campania region: S. Martino in Naples, S. Lorenzo in Padula (Salerno) and S. Giacomo, in Capri. All the texts, the architectural surveys and the audio-video recordings, in other words, all the digital resources that we have and will collect and that we describe in the next sections, concern with these wonderful sites.

2 The Challenge

An interesting aspect of the CHROME project is to tackle some methodological and technological challenges.

A first challenge regards the role of gatekeepers in shaping visitors' experience. In fact, the communication in museums is considered an important issue even if museum specialists have been reproached to not do enough in this field (Antinucci, 2014), with some exceptions. Many advancements have been obtained concerning the attempt to understand museum visitors needs and to look for new ways of communication to improve the experience of visiting museums. Investigations about visitors psychological approach (Dufresne-Tassé C. & Lefebvre A., 1995) helped museologists to develop possible methods not only to exhibit artefacts but also to give them sense, providing further explanations. So museum experts may better know visitors, and they are ready to be helped by technology (Cataldo L., 2011).

Moreover, another important aim regards the extraction of concepts and expressive forms from texts. Natural Language Processing technologies are crucial in the process of converting textual documents into knowledge resources. New techniques for the automatic acquisition of linguistic knowledge from texts are needed. Terminology extraction is a central field of research for a number of applications, such as Ontology Learning and Text Mining. Different methodologies have been proposed so far to automatically extract domain terminology from texts. Term extraction systems make use of various degrees of linguistic filtering and of statistical measures ranging from raw frequency to Information Retrieval measures such as TF-IDF (Salton et al.,

1988), up to more sophisticated methods such as the C-NC Value method (Frantzi et al., 1999) or contrastive approach (Bonin et al., 2010).

Another important issue we are going to manage is the analysis of social behaviors in dissemination contexts. The specificities of guided tours have been investigated in (Mondada, 2013), who studies the distribution of knowledge among guides. This stresses the need to adapt to different people during visits; while the relevance of a user model is pointed out by literature in gesture and Conversational Analysis. Concerning the use of words and iconic gestures in didactic explanations to children and expert and novice adults, their adaptation to the Speaker's Recipient Design and their efficacy for comprehension, (Campisi & Özyürek, 2013) show that people use more words when addressing to adults, but wider and more informative gestures for children. Also, precision was defined as providing details on the

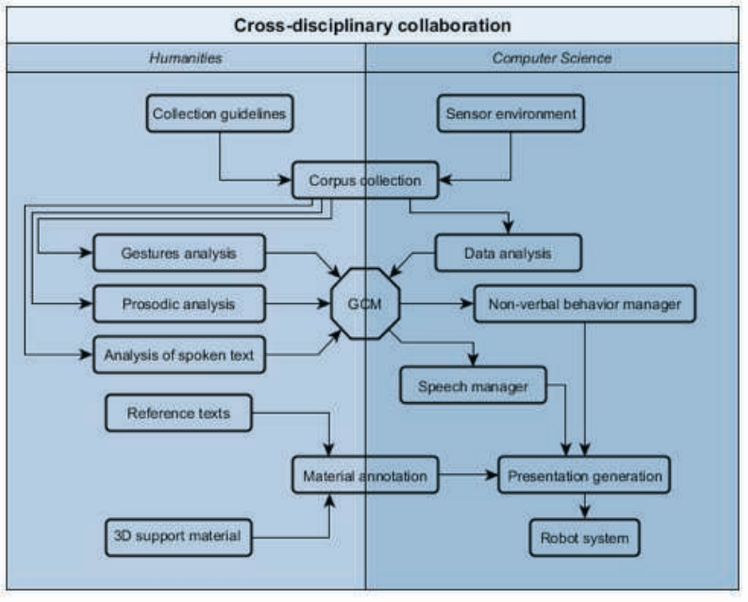


Fig. 1 The CHROME interdisciplinary chart

topic of one's discourse (Vincze et al., 2014), while vagueness is how blurred are the boundaries of one's ideas or discourse.

Spoken text analysis and, prosodic analysis and synthesis will also be addressed. Advanced use of parametric speech synthesis, such as focus/prominence generation by prosodic modification or expressive prosody modelling, has been tested in some research projects (i.e. ALIZ-E). Pushing forward prosodic analysis on gatekeepers' performance can improve the knowledge needed to synthesize natural specialized speech.

Finally, the technologies to mediate the access to digital cultural heritage will be considered. In order to dynamically assemble and present narratives, a formalism to represent different aspects of cultural stories (i.e. (Mele & Sorgente, 2013)) as reported by gatekeepers is necessary. By providing semantically annotated multimedia materials and contents obtained collecting a documental basis, it is possible to use mash-up techniques to dynamically assemble contents and synchronize them with the available media.

3 CHROME methodology

CHROME is a cross-disciplinary project focused on combining computational linguistics and behavior analysis methods with expertise in museology to formalise computational models of gatekeepers (see Fig. 1). The main result of this research will be the Gatekeeper Computational Model (GCM) to generate engaging presentations of cultural heritage. The project is organized in three main phases. The data collection phase foresees recording of gatekeepers presenting cultural contents and surveying activities to collect reference texts and annotated 3D models. During data analysis, these resources will be annotated and examined to obtain the GCM. Activities will compare oral expressions with expressions found in texts to automatically select fragments that can compose the final presentation together with gestures and prosody synthesis. 3D models annotation will allow to connect presentation to automatic selection of auxiliary material. Demonstrator implementation will

serve for the validation of the GCM, to disseminate the research results and estimate the impact of the approach in a real environment.

The methodology proposed in the CHROME project targets the following objectives:

- O1. Provide reference datasets to study human-human interaction during the presentation of cultural heritage.
- O2. Survey written contents for cultural heritage dissemination and compare these with the multimodal materials collected in the framework of the CHROME project.
- O3. Provide a reference Gatekeeper Computational Model (GCM) to support development of artificial agents mimicking the ability of expert guides to select and organize contents and applying proper verbal and non-verbal behaviour
- O4. Evaluate the impact of dissemination oriented, multimodal behavioral models on the capability of artificial agents to simplify access to digital cultural heritage and attract visitors in cultural sites

4 The present status

At the time we are writing this paper (July 2018) we are at month 16 of 36. Up to now we have collected and analysed many data on Campania Charterhouses: texts, audio, video and 3D reconstructions.

4.1 Charterhouses Text

For the three Campania Charterhouses (S. Martino, S. Lorenzo and S. Giacomo), we have collected 102 texts that belong to different document types. In particular, such texts are divided among the following categories: Scientific texts; Specialized catalogues; Dissemination catalogues; Specialized guides; Certified web material; Dissemination kits.

4.2 Textual Analysis

Starting from these texts, some lexical and semantic analyses have already been conducted on part of them. The main ones concerned: *i*) Domain vocabulary extraction; *ii*) Event annotation: some texts are annotated added semantic information with respect to reference formalism event based. In particular, the formalism adopted is CSWL (Cultural Story Web Language) (Sorgente et al., 2016). The purpose of this approach is to have a semantic level that will allow us to define an information retrieval not only based on text search; *iii*) AAT concepts recognition: the Art & Architecture Thesaurus (AAT) (Getty, 2018) is a structured vocabulary containing around 40,000 concepts and descriptions related to fine art, architecture, decorative arts, archival materials and material culture. In this step the aim is to link the concepts inside charterhouses texts to such vocabulary.

4.3 Digital photogrammetry

The architects group have completed the activity of aerial photogrammetry digital survey performed by UAV and laser scanner on the 3 main charterhouses buildings and on many interiors.

4.4 Video recording of touristic guide

Three of four touristic guides have been video recorded during tours in the S. Martino Charterhouse while describing the artistic features, and each one is followed by a public of four visitors. Cameras are pointed on the guide and on the public, speech sounds are recorded with three microphones, one headset worn by the guide and two on field at about one meter equidistant from the guide and pointing to the visitors, too. Speech analyses on these material consists of:

- Orthographic level: Transcription of words, pauses, filled pauses, false starts;
- Phonetic level: Phonetic transcription and annotation of coarticulation phenomena, Speech quality analysis;
- Syllabic level: Annotation of syllables, Speech fluency and speech rate analysis;
- Intonation level: Pitch movements in relationship with the segmental level, Emphasizing patterns, speech style.
- Textual level: analysis of sentences, text structure, and communicative goals.
- Multimodal behavior level: annotation of gestures, face and gaze, including physical description, semantic analysis, classification in terms of textual, emotional and interactional functions.

The tool chosen for annotating the speech and video material is ELAN[1] . In each video portion the guide's gestures and body communication will be annotated in terms of the communicative functions they serve. Thus the annotation will allow to distinguish the styles of the guides: e.g. a very "technical" guide will use gestures and body communication more frequently aimed at describing the artwork or the author, while a "friendly" guide's body behaviors will be often aimed at creating syntony with tourists.

[1] https://tla.mpi.nl/tools/tla-tools/elan/

5 Summarizing

CHROME aims at formalizing data collection and annotation paradigms for architectural heritage, in particular the annotation regards texts, video, audio and gestures. From the annotated data, we will: *i*) perform correlation analysis to identify cross-domain patterns and link them to communicative goals; *ii*) describe how an expert presenter relates to the physical environment while she describing it; *iii*) identify which communicative strategies can be mimicked by an artificial agent with the available technology. Possible domains of simulation will the deictic and iconic gestures, face and gaze behaviour; *iv*) implement a final demonstrator adopting the formalized strategies to generate dynamic presentations for the attending visitors.

6 Aknowledgments

This work is funded by the Italian PRIN project Cultural Heritage Resources Orienting Multimodal Experience (CHROME) #B52F15000450001.

Reference

Antinucci F. (2014). Comunicare nel museo. Laterza Milano

Bonin F., Dell'Orletta F., Montemagni S., Venturi G. (2010). *A Contrastive Approach to Multi-word Extraction from Domain-specific Corpora*. In: LREC'10 – Seventh International Conference on Language Resources and Evaluation (Valletta, Malta, 17-23 May 2010). Proceedings, pp. 3222 – 3229.

Campisi E. and Ozyürek, A. (2013). Iconicity as a communicative strategy: Recipient design in multimodal demonstrations for adults and children. Journal of Pragmatics (47), pp. 14-27

Cataldo L. (2011). Dal Museum Theatre al Digital Storytelling. Franco Angeli Milano

Dufresne-Tassé C., Lefebvre A. (1995). Psychologie du visiteur du musée. Hurtubise Montréal

Mele, F., Sorgente, A. (2013). OntoTimeFL – A Formalism for Temporal Annotation and Reasoning for Natural Language Text. New Challenges in Distributed Information Filtering and Retrieval, Studies in Computational Intelligence 439, pp. 151-170

Mondada, L. (2013). Displaying, contesting and negotiating epistemic authority in social interaction: Descriptions and questions in guided visits. Discourse Studies 15, pp. 597-626

Frantzi, K., Ananiadou, S. (1999). The C–value / NC Value domain independent method for multi–word term extraction.

Getty AAT: About the AAT. http://www.getty.edu/research/tools/vocabularies/aat/. Accessed April 2018

Sorgente A., Calabrese A., Coda G., Vanacore P., and Mele F. (2016). Building multimedia dialogues annotating heterogeneous resources. In Artificial Intelligence for Cultural Heritage, chapter 3, pages 49–82. Cambridge Scholars Publishing.

Salton G., Buckley, C. (1988). Term–Weighting Approaches in Automatic Text Retrieval. Information Processing and Management, 24(5), pp. 513-523

Vincze L., Poggi I., D'Errico F. (2014). Precision in Gestures and Words. Ricerche di Pedagogia e Didattica – Journal of Theories and Research in Education 9, 1. Communicating certainty and uncertainty: Multidisciplinary perspectives on epistemicity in everyday life

Italian in the Trenches: Linguistic Annotation and Analysis of Texts of the Great War

Irene De Felice•, Felice Dell'Orletta◇, Giulia Venturi◇, Alessandro Lenci•, Simonetta Montemagni◇
• University of Pisa, CoLing Lab
irene_def@yahoo.it, alessandro.lenci@unipi.it
◇Istituto di Linguistica Computazionale "A. Zampolli", ItaliaNLP Lab
{felice.dellorletta, giulia.venturi, simonetta.montemagni}@ilc.cnr.it

Abstract

English. The paper illustrates the design and development of a textual corpus representative of the historical variants of Italian during the Great War, which was enriched with linguistic (lemmatization and pos-tagging) and meta-linguistic annotation. The corpus, after a manual revision of the linguistic annotation, was used for specializing existing NLP tools to process historical texts with promising results.

Italiano. *L'articolo illustra la progettazione e la costruzione di un corpus rappresentativo delle varietà di italiano in uso durante la prima Guerra Mondiale, annotato con dati linguistici (lemmatizzazione, analisi morfo-sintattica) e meta-linguistici. Il corpus, a seguito della revisione manuale dell'annotazione linguistica, è stato utilizzato per l'adattamento degli strumenti NLP esistenti, con risultati promettenti.*

1 Introduction

World War I (WWI) represents a crucial period in the history of Italian. In fact, De Mauro (1963) claimed that Italian as a national language was born in the trenches of the Great War. Since masses of men from different regions of the peninsula were forced to live together for months in the trenches and behind the lines, and were forced to use Italian as the main communicative medium instead of regional dialects, WWI produced a decisive step forward in the process leading to the linguistic unification of Italy.

The project *Voci della Grande Guerra* (*VGG*)[1] provides scholars with a new text corpus to investigate the structure and different varieties of Italian at the time of the Great War. The corpus includes a selection of texts representative of different textual genres and registers, including popular Italian. All texts have been automatically annotated with state-of-the-art NLP tools. A large subset of the corpus has then been manually corrected and enriched with metadata to classify a broad range of phenomena relevant for the study of the linguistic features of early XX century Italian. These characteristics make the *VGG* corpus unique in the very limited panorama of existing Italian historical corpora, among which it is worth pointing out the corpus dell'*Opera del Vocabolario Italiano* (OVI), the *DiaCORIS* corpus (Onelli *et al.*, 2006), the *MIDIA* corpus (Gaeta *et al.*, 2013), and the *Letteratura italiana Zanichelli* (LIZ). Moreover, the developed *VGG* corpus was used in an interesting case-study for the application and adaptation of NLP tools to process historical texts. The aim of this paper is to present the results of the annotation and linguistic analysis of the *VGG* corpus.

2 The Corpus *Voci della Grande Guerra*

The *VGG* corpus consists of 91 texts (ca. 1M tokens) that were written in Italian in the period of the World War I or shortly afterwards (most of them date back to the years 1915-1919). The texts were selected by historians and linguists in order to represent the 'polyphony'of the different voices of people who were affected by World War I. The corpus is balanced with respect to genre, style, and authors' profession: it collects discourses, reports and diaries of politicians and military chiefs; letters written by men and women, soldiers and civilians; literary works of intellectuals, poets, and philosophers; writings of journalists and lawyers.

Most documents existed only in printed form and were scanned and digitized with OCR tools. Once digitized, the documents were codified in the TEI-XML standard format. A significant part of the corpus of about 650,000 tokens, for which the

[1] http://www.vocidellagrandeguerra.it/

output of the OCR was manually corrected line–by–line with a correction tool specially designed for this purpose, constitutes our textual gold standard (Boschetti *et al.*, 2018).[2]

As a second step, documents were exported to be processed with NLP tools (cf. Section 3). Automatic linguistic annotation has been manually checked and corrected for more than 500,000 tokens for sentence splitting, tokenization, and lemmatization. For one fifth of this revised part of the corpus (ca. 103,000 tokens), manual revision has also targeted PoS tagging and morphological analysis. The revised documents belong to different genres and styles (see Table 1).

3 Method

The annotation methodology we have employed for the construction of the *VGG* corpus was articulated in the following steps:

1. the whole *VGG* corpus was automatically annotated using *UDPipe*, a trainable pipeline for tokenization, pos-tagging, lemmatization and dependency parsing with a transition based parser based on a non-recurrent neural network, with just one hidden layer, with locally normalized scores (Straka and Straková, 2017). The pipeline was trained on the Italian Universal Dependency Treebank (IUDT), version 2.0 (Bosco *et al.*, 2013);

2. the linguistic annotation of the *VGG* subcorpus reported in Table 1 was manually revised and whenever needed corrected. As fully described in Section 4, it was also enriched with metalinguistic information aimed to highlight features characterizing the variety of Italian used in the historical period considered. Correction was performed with a UD-compliant annotation tool specifically designed for the project.

3. the manually revised sub-corpus was used to retrain the automatic linguistic annotation pipeline in order to improve the performance of the automatic analysis tools.

4 Manual revision and meta-linguistic annotation

The first phase of automatic linguistic analysis performed on the *VGG* corpus (see Section 3) did not prove to be sufficient to achieve an accurate annotation of the texts, for two main reasons. First of all, the *VGG* corpus represents a historical variety of language, therefore obsolete forms are frequently found at both the lexical and the morphological level. Moreover, the documents feature an impressive degree of linguistic variation, which reflects the level of education of the writers, the style and register of texts (which in turn depend on their targeted purposes and audience, and on the particular social settings in which they were written), and the regional diversification of the Italian language in the years of the WWI (which was still largely permeated with dialectal features). Current NLP tools, trained on texts representative of standard, contemporary Italian (cf. Section 5), are not able to handle such a huge linguistic variation (see the performance reported in Table 2). Therefore, we performed a manual revision of the automatic annotation on a gold subsection of the corpus and enriched it with additional data, in order to retrain and improve the language model.

4.1 Manual revision

Automatic annotation was manually checked and corrected for more than 500k tokens for sentence splitting, tokenization, lemmatization, and partly also for PoS tagging and morphological analysis (cf. Table 1). This operation allowed us to individuate the most relevant features of the *VGG* corpus that pose critical difficulties to automatic annotation, as briefly illustrated in what follows.

Major issues with tokenization:

1. *Pronominal clitics attached to verbs*. Although pronominal clitics regularly attach to verbs in Italian under particular conditions, some combinations (e.g., *abbiti*, *siasi*) are very rare in contemporary Italian and linguistic tools often fail in segmenting and analyzing them correctly. Such forms were manually identified and splitted (*abbi+ti*, *sia+si*).

2. *Hyposegmentation*. When two or more words appear erroneously unsegmented (as it frequently happens in texts written by uneducated people), they were manually split and analyzed separately (*sela=se+la*, *inmente=in+mente*), similarly to the tool that automatically splits articulated prepositions and verbs with clitics.

[2] We plan to extend the manual revision of the output of the OCR, which is still ongoing, to approximately 1M tokens.

Text genre	Tok. + Lemm.	Tok. + Lemm. + PoS
Diary (Gadda, Martini, Sonnino)	43,419	49,868
Discourse (D'Annunzio, Morgari, Salandra, Salvemini, Treves, Turati; dichiarazioni del Partito Socialista)	44,942	7,792
Essay (Croce, Gemelli, Gentile)	8,352	9,524
Letters (Fontana, Monteleone, Monti, Procacci, Raviele)	89,938	5,310
Memoir (Cadorna, Jahier, Monelli, Prezzolini, Soffici)	134,874	22,938
Report (Comitati Segreti della Camera dei Deputati)	75,549	7,573
Tot.	397,074	103,005

Table 1: For each genre, number of tokens manually revised (for tokenization and lemmatization only, or also for PoS and morphological features).

Major issues with lemmatization:

1. *Rare terms*. The VGG corpus is rich with terms that are rare or old-fashoned in standard contemporary Italian (e.g., *costí*, *ingramagliare*), and that for this reason are rarely analyzed correctly. For such forms, the correct annotation was manually entered.

2. *Variants of lemmas*. Automatic tools often fail in lemmatizing a word correctly, when it does not refer to a standard lemma of contemporary Italian, but to one of its possible variants (e.g., *comperare* for *comprare*, *spedale* for *ospedale*). In such cases, both the standard and the variant lemma are manually annotated (359 different variant lemmas were found so far, for a total of 1361 occurrences).

3. *Misspellings*. In informal texts, words are often lemmatized incorrectly because they are wrongly spelled. For instance, *o* and *anno* may be the misspelled inflected forms of the verb *avere* (*ho*, *hanno*), and not just the conjunction *o* and the noun *anno*. In these cases, the correct linguistic annotation was added.

Major issues with morphological analysis:

1. *Variants in inflectional morphology*. Words that present rare or old-fashioned morphological formations (e.g., 3pl. pres. subj. *sieno* for standard It. *siano*; 2sg. fut. ind. *anderai* for standard It. *andrai*) in most cases are wrongly analyzed by the automatic tool and were therefore manually corrected.

4.2 Metalinguistic annotation

During the manual revision of the annotation (conducted on more than 500k tokens), an additional level of metalinguistic annotation was added. Words that can be considered as 'marked' with respect to standard contemporary Italian, and that are explicitly signaled as such in dictionaries (e.g., as literary or archaic forms), were manually identified and classified according to how they are labeled in the lexical resources consulted (*Dizionario De Mauro*, *Dizionario Hoepli*, *Dizionario Sabatini-Coletti*, and *Vocabolario Treccani*). We adopted the following labels:
dial: for forms classified as dialectal (e.g. *batajun*, *preive*; tot. 1,536 annotations).[3]
lit: for forms classified as literary or poetic (e.g. *pelago*, *nocumento*; tot. 1,046 annotations).
uncomm: for forms classified as rare and unfrequent (e.g. *impinguire*, *sconcordia*; tot. 891 annotations).
ant: for forms classified as obsolete or archaic (e.g. *imperocché*, *tardanza*; tot. 474 annotations).
reg: for forms classified as regional, i.e. typical of a regional variety of Italian (e.g. *cocuzza*, *mencio*; tot. 232 annotations).
pop: for forms classified as popular or vulgar (e.g. *pisciare*, *minchione*; tot. 134 annotations).

These labels (tot. 4,313 annotations) can be associated: (i) to a lemma (e.g. *tardanza*, *pelago*); (ii) to a variant lemma, in which case we add to the label the feature **var** (e.g., *imaginazione*, 'lit. var.' of the standard lemma *immaginazione*); (iii) to a single inflected form marked at the morphological level, in which case we add to the label the feature **morph** (e.g., *dieno*, 'morph. ant.' form of the 3pl. pres. subj. of the verb *dare*). Moreover, the same form may also receive two labels (e.g., *periglioso*, marked as 'ant./lit.').

Finally, misspelled or wrongly segmented forms (e.g., *Cavur* for *Cavour*, *cuatro* for *quat-*

[3]Not all dialectal forms are listed in Italian dictionaries. Nevertheless, they can be confidently identified in texts, since dialectal elements mostly appear in sequences, for instance in proverbs, songs, or poems. Moreover, authors often enclose dialectal forms in double quotation marks, or write them in italics.

tro, *inmente* for *in mente*) were also marked with a specific label: **err** (tot. 5,251 annotations).

It is evident that the metalinguistic annotation of marked forms is particularly relevant from a (socio-)linguistic point of view, since it offers an insight into the different dimensions of linguistic variation of the Italian language of the years of the WWI, from a diachronic, diatopic, diaphasic and diastratic points of view.

5 Automatic Linguistic Annotation

Automatic linguistic analysis of historical texts is a complicated venture. As reported in Piotrowski (2012), the main challenge is high variation on all levels both across and within texts, for instance due to the absence of standardized spelling, the occurrence of historical variants of words as well as peculiar syntactic structures. For these reasons, contemporary tools for linguistic analysis are generally not suitable for processing historical texts. This is the problem we faced in the project: as reported in Section 4 the texts of the VGG corpus differ in many respects from modern Italian.

Table 2 reports the performance recorded for the different levels of automatic linguistic annotation of the *VGG* corpus, using general and specialized language models. We tested the whole annotation pipeline on two test sets representative of two very different textual genres, i.e. discourses and letters, in order to assess the impact of different language varieties on the performance of the analysis tools.

We first trained UDPipe on IUDT v2.0: a significantly high drop of accuracy can be observed with respect to the state-of-the-art performance on modern Italian (Straka and Straková, 2017). In particular, for the letters collected by Monteleone very low performance is reported at all levels of analysis. This is mainly due to the features of this language variety: the letters were often written by uneducated people, they are characterized by a colloquial style, reminiscent of spoken language that is quite different from the typology of texts used for training. The split of sentences is the least accurate level of analysis: a non canonical use of punctuation both in Salandra's discourses and in the corpus of letters can be the main cause. On the other hand, token segmentation resulted to be less negatively affected in both cases.

Once the sub-corpus of ∼100k manually revised tokens was available, which included documents representative of the different textual genres considered, it was combined with the IUDT training data to retrain UDPipe. As expected, a general improvement was achieved at all analysis levels. For the two textual genres chosen for testing, the highest improvement turned out to be concerned with lemmatization. As discussed in Section 4, the *VGG* corpus contains several rare lexical items, old lemma variants, misspellings due to uneducated or informal use of language. The manual correction of the lemma helped to improve lemmatization and, similarly, PoS tagging.

6 Conclusions and current developments

Voices of the Great War is the first large corpus of documents in Italian dating back to the period of WWI. This corpus differs from other existing resources because it gives account of the wide range of varieties in which Italian was articulated in the years of WWI, namely from a diastratic (educated vs. uneducated writers), diaphasic (low/informal vs. high/formal registers) and diatopic (regional varieties, dialects) points of view. The linguistic variety subsumed in the corpus posits a number of challenges for current NLP tools, which are trained on texts representative of standard contemporary Italian. In this paper, we showed how we faced such challenges, by developing a more efficient model for the analysis of Italian texts of the period of WWI.

For approximately 20,000 tokens of the manually revised part of the corpus, we are building a syntactic annotation level performed according to the Universal Dependency scheme, which will constitute the first small treebank for historical Italian.

At the end of the project, the texts not covered by copyright will be freely dowloadable together with their annotations. The other texts will instead be browsable online with a dedicated interface.

References

Federico Boschetti, Andrea Cimino, Felice Dell'Orletta, Gianluca E. Lebani, Lucia Passaro, Paolo Picchi, Giulia Venturi, Simonetta Montemagni, and Alessandro Lenci. 2014. Computational analysis of historical documents: An application to italian war bulletins in WWI and WWII. In *Proceedings of the LREC 2014 Workshop on Language resources and technologies for processing and linking historical documents and archives - Deploying Linked Open Data in Cultural*

Test	Sentence Spitting	Tokenization	Lemmatization	PoS Tagging
Training: IUDT				
IUDT	**97.1%**	**99.8%**	**97.03%**	**97.02%**
Training: IUDT				
Discourses (Salandra-1922)	82.20% (-14.90)	99.53% (-0.27)	85.80% (-11.23)	83.36% (-13.66)
Letters (Monteleone)	65.58% (-31.52)	99.15% (-0.65)	83.05% (-13.98)	79.45% (-17.57)
Training: IUDT+VGG				
Discourses (Salandra-1922)	**92.46% (+10.26)**	99.74% (+0.21)	**95.20% (+9.40)**	90.82% (+7.46)
Letters (Monteleone)	69.46% (+3.88)	99.69% (+0.54)	**90.80% (+7.75)**	84.93% (+4.98)

Table 2: Comparison of F-scores in different annotation tasks using IUDT (*IUDT Training*) and combining out- and in-domain training data (*IUDT+VGG Training*) on different test sets. In parenthesis the relative improvement or drop of accuracy with respect to Straka and Straková (2017).

Heritage (LRT4HDA 2014, Reykjavik, Iceland), 70–75.

Cristina Bosco, Simonetta Montemagni and Maria Simi. 2013. Converting Italian Treebanks: Towards an Italian Stanford Dependency Treebank". *Proceedings of the ACL Linguistic Annotation Workshop & Interoperability with Discourse*, Sofia, Bulgaria.

Federico Boschetti, Michele Di Giorgio and Nicola Labanca. 2018. Bisogna farli parlare: la formazione di un corpus di Voci della Grande Guerra e il comma 22. In Mirko Volpi (ed.), *Atti del primo convegno "Voci della Grande Guerra"*, Firenze, Accademia della Crusca, pp. 14-31.

Felice Dell'Orletta and Giulia Venturi. 2016. ULISSE: una strategia di adattamento al dominio per l'annotazione sintattica automatica. In Edoardo Maria Ponti and Marco Budassi (eds.), *Compter parler soigner: tra linguistica e intelligenza artificiale. Atti del convegno 15-17 dicembre 2014*, 55-79. Pavia University Press, Pavia.

Tullio De Mauro. 1963. *Storia linguistica dell'Italia unita*. Laterza, Bari.

Dizionario De Mauro = *Il Nuovo De Mauro*. Available online at: `https://dizionario.internazionale.it`.

Dizionario Hoepli = Aldo Gabrielli. *Grande Dizionario Italiano Hoepli*. Available online at: `http://dizionari.hoepli.it`.

Dizionario Sabatini-Coletti = Francesco Sabatini and Vittorio Coletti. *Il Sabatini Coletti Dizionario della lingua italiana*. Available online at: `http://dizionari.corriere.it/`.

L. Gaeta, C. Iacobini, D. Ricca, M. Angster, A. De Rosa, G. Schirato. 2013. MIDIA: a balanced diachronic corpus of Italian. 21st International Conference on Historical Linguistics, Oslo.

Alessandro Lenci, Nicola Labanca, Claudio Marazzini, Simonetta Montemagni. 2016. Voci della Grande Guerra: An Annotated Corpus of Italian Texts on World War I. *Italian Journal of Computational Linguistics*, 2(2):101–108.

Corinna Onelli, Domenico Proietti, Corrado Seidenari, Fabio Tamburini. 2006. The DiaCORIS project: a diachronic corpus of written Italian. Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC 2006, Genoa, Italy, May 22-28, 2006, pp. 1212–1215.

Lucia Passaro and Alessandro Lenci. 2014. "Il Piave mormorava...": Recognizing locations and other named entities in Italian texts on the great war. In Roberto Basili, Alessandro Lenci, and Bernardo Magnini (eds.), *Proceedings of the First Italian Conference on Computational Linguistics*, 286–290. Pisa University Press, Pisa.

Michael Piotrowski. 2012. Natural Language Processing for Historical Texts. Synthesis Lectures on Human Language Technologies, Morgan & Claypool Publishers.

Milan Straka and Jana Straková. 2017. Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe. *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, Vancouver, Canada, pp. 88–99.

Vocabolario Treccani = *Il Vocabolario Treccani*. Available online at: `http://www.treccani.it/vocabolario/`.

Lexicon and Syntax: Complexity across Genres and Language Varieties

Pietro dell'Oglio•, Dominique Brunato◇, Felice Dell'Orletta◇
• University of Pisa
pietrodelloglio@live.it
◇Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC–CNR)
ItaliaNLP Lab - *www.italianlp.it*
{dominique.brunato, felice.dellorletta}@ilc.cnr.it

Abstract

English. This paper presents first results of an ongoing work to investigate the interplay between lexical complexity and syntactic complexity with respect to nominal lexicon and how it is affected by textual genre and level of linguistic complexity within genre. A cross-genre analysis is carried out for the Italian language using multi–leveled linguistic features automatically extracted from dependency parsed corpora.

Italiano. *Questo articolo presenta i primi risultati di un lavoro in corso volto a indagare la relazione tra complessità lessicale e complessità sintattica rispetto al lessico nominale e in che modo sia influenzata dal genere testuale e dal livello di complessità linguistica interno al genere. Un'analisi comparativa su più generi è condotta per la lingua italiana usando caratteristiche linguistiche multilivello estratte automaticamente da corpora annotati fino alla sintassi a dipendenze.*

1 Introduction

Linguistic complexity is a multifaceted notion which has been addressed from different perspectives. One established dichotomy distinguishes a "global" vs a "local" perspective, where the former considers the complexity of the language as a whole and the latter focuses on complexity within each sub-domains, i.e. phonology, morphology, syntax, discourse (Miestamo, 2008). While measuring global complexity is a very ambitious and probably hopeless endeavor, measuring local complexities is perceived as a more doable task (Kortmann and Szmrecsanyi, 2012). The level of complexity within each subdomains indeed has been formalized in terms of distinct parameters that capture either internal properties of the language (in the "absolute" notion of complexity) or phenomena correlating to processing difficulties from the language user's viewpoint (in the "relative" notion of complexity) (Miestamo, 2008). For instance, complexity at lexical level has been computed in terms of *length* (measured in characters or syllables), of *frequency* either of the whole surface word (Randall and Wayne, 1988; Chiari and De Mauro, 2014) or of its internal components (see e.g. the *root frequency effect* (Burani, 2006)), *ambiguity* and *familiarity*, among others. At syntactic level, much attention has been paid on canonicity effects due to word order variation (Diessel, 2005; Hawkins, 1994; Futrell et al., 2015), as well as on long-distance dependencies (Gibson, 1998; Gibson, 2000) proving their effect on a wide range of psycholinguistic phenomena, such as the subject/object relative clauses asymmetry or the garden path effect in main verb/reduced–relative ambiguities.

An interesting question addressed by recent corpus-driven research is how language complexity is affected by textual genre. At syntactic level, the study by Liu (2017) on ten genres taken from the British National Corpus showed that genre-specific stylistic factors have an influence on the distribution of dependency distances and dependency direction. Similarly for Italian, Brunato and Dell'Orletta (2017) investigated the influence of genre, and level of complexity within genre, on a range of factors of syntactic complexity automatically computed from dependency-parsed corpora. Inspired by that work, we also intend to analyze the effect of genre on linguistic complexity. However, unlike the dominant local approach, where each subdomain is typically studied in isolation, our contribution intends to address the interrelation between different levels, i.e. lexicon and syntax. Specifically, we investigate the fol-

lowing questions:

- to what extent is lexical complexity influenced by genre?
- to what extent is lexical complexity influenced by the level of complexity within the same genre?
- is there a correlation between lexical complexity and syntactic complexity? Does it vary according to genre and level of complexity within the same genre?

To answer these questions, we conducted an in-depth analysis for the Italian language based on automatically dependency parsed corpora aimed at assessing i) the distribution of simple and complex nominal lexicon in different genres and different language varieties for the same genre ii) the syntactic role bears by "simple" and "complex" nouns characterizing each corpus iii) the correlation between "simple" and "complex" nouns with features of complexity underlying the syntactic structure in which they occur.

In what follows we first describe the corpora considered in this study. We then illustrate how lexical and syntactic complexity have been formalized. In Section 4 we discuss some preliminary findings obtained from the comparative investigation across corpora.

2 The Corpora

Four genres were considered in this study: Journalism, Scientific prose, Educational writing and Narrative. For each genre, we chose two corpora, selected to be representative of a complex and of a simple language variety for that genre. The level of complexity was established according to the expected target audience.

The Journalistic corpora are *Repubblica* (Rep) for the complex variety, and *Due Parole* (2Par) for the simple one. Rep is a corpus of 232,908 tokens and it is made of all articles published between 2000 and 2005 on the newspaper of the same name; 2Par contains 322 articles taken from the easy-to-read magazine *Due Parole*[1], for a total of about 73K tokens.

The corpora representative of Scientific writing are *Scientific articles* (ScientArt) for the complex language variety, and *Wikipedia articles* (WikiArt) for the simple one. The former is made of 84 documents (471,969 tokens) covering various topics on scientific literature. The latter is made of 293 documents (about 205K tokens) extracted from the Italian web portal "Ecology and Environment" of Wikipedia.

For the Educational writing corpora we relied on two collections of school textbooks: the 'complex' one (EduAdu) contains 70 texts (48,103 tokens) targeting high school students, the 'simple' one (EduChi) a sample of 127 texts (48,036 tokens) targeting primary school students.

Finally, the Narrative corpora are composed by the original versions of *Terence* and *Teacher* (TTorig), for the complex pole, and the correspondent simplified versions for the simple pole. Terence, which is named after the EU Terence Project[2], is made of 32 documents, covering short novels for children. Teacher contains 24 documents extracted from web sites dedicated to educational resources for teachers. All *Terence* and *Teacher* texts have a simpler version (TTsemp), which is the result of a manual simplification process as described by Brunato and Dell'Orletta (2017).

All corpora were automatically tagged by the part-of-speech tagger described in (Dell'Orletta, 2009) and dependency parsed by the DeSR parser described in (Attardi et al., 2009).

3 Features of Linguistic Complexity

3.1 Assessment of Lexical Complexity

For each corpus we extracted all lemmas tagged as nouns, without considering proper nouns, and we classified them as 'simple' vs 'complex' nouns. Such a distinction was established according to their frequency, which is one of the most used parameter to assess the complexity of vocabulary (see Section 1). Frequency was here computed with respect to a reference corpus, i.e. ItWac (Baroni et al., 2009), which was chosen since this is the biggest corpus available for standard Italian thus offering a reliable resource to evaluate word frequency on a large-scale. After ranking all nouns for frequency, we pruned those with a frequency value ≤ 3 and we kept the first quarter of nouns as representative of the sample of *simple* nouns and the last quarter as representative of the sample of *complex* nouns for each corpus.

[1] www.dueparole.it

[2] www.terenceproject.eu

3.2 Assessment of Syntactic Complexity

To investigate our main research questions, that is how lexical complexity affects syntactic complexity and the possible influence of genre and language variety on this relationship, we focused on a set of features automatically extracted from the sentence parse tree. These features were chosen since they are acknowledged to be predictors of phenomena of structural complexity, as demonstrated by their use in different scenarios, such as the assessment of learners' language development or the level of text readability (e.g. (Collins-Thompson, 2014; Cimino et al., 2013; Dell'Orletta et al., 2014)).

For each corpus, all the considered features were computed for all occurring nouns, for the subset of *complex* nouns and for the subset of *simple* nouns. Specifically, we focused on the following ones:

- The linear distance (in terms of tokens) separating the noun from its syntactic head (*HeadDistance* in all following Tables)
- The hierarchical distance (in terms of dependency arcs) separating the noun from the root of the tree (*RootDistance*)
- The average number of children per noun (*AvgChildren*)
- The average number of siblings per noun (*AvgSibling)*

4 Discussion

To have a first insight into the effect of genre and language variety on the interplay between lexical and syntactic complexity, we compared the main syntactic roles that nouns play in the sentence by calculating the frequency of all dependency types linking a noun to its head. This is shown in Figure 1, which reports the percentage distribution of typed dependency relationships linking a noun to its syntactic head across all corpora. For each corpus there are three columns: the first one considers data for all nouns of each corpus without any complexity label, the second one only data for the *simple* noun subset and the last one only data for the *complex* noun subset.

It can be noted that the distribution of nouns used as prepositional complements (prep) is the higher one across all corpora although with differences ranging from the lowest percentage (35.5%) in the 'easy' version of the narrative corpus (i.e. *TTsemp*) to the highest one (49.9%) in ScientArt (i.e. the complex language variety for the scientific writing genre). The syntactic role of prepositional complement is especially played by *simple* nouns compared to *complex* nouns. This is particularly evident in *ScientArt* and *Repubblica*, where the difference between *simple* and *complex* nouns occurring as prepositional complements is equal respectively to 20 and 15 percentage points. Conversely, *complex* nouns are more widely used as modifiers than *simple* nouns, especially in *Repubblica*. The percentage of nouns occurring in the subject and object position is less than 20% in all corpora. Interestingly, the higher occurrence of nominal subjects is attested in *DueParole* and *ChildEdu* (14.1 and 16, respectively). This might suggest that simpler language varieties, independently from genre, make more use of explicit subjects than implicit or pronominal ones. Besides, the likelihood of a noun to be simple or complex does not particularly affect the overall presence of nominal subjects, unless for *ScientArt* and *Rep* which both show a higher percentage of *simple* nouns in the subject position.

A deeper understanding of the relationship between lexical and syntactic complexity was provided by the investigation of the syntactic features described in Section 3.2. Table 1 shows the average value of the monitored features with respect to all nouns (All), to the subset of *complex* nouns (Comp) and to the subset of *simple* nouns (Simp) extracted from all corpora. We assessed whether the variation between these feature values was statistically significant in a three different comparative scenarios: i) between the two corpora of the same genre, ii) between the complex corpora of each different genre and ii) between the simple corpora of each different genre. Table 2 shows linguistic features varying significantly for all the considered comparisons according to the Wilcoxon rank-sum test, a non parametric statistical test for two independent samples (Wild, 1997).

If we compare the two language varieties within each genre, it can be seen, for instance, that nouns are hierarchically more distant from the root in the complex than in the simple version. Such a variation, which is highly significant for all genres, affects more the Journalistic genre (*DuePa-*

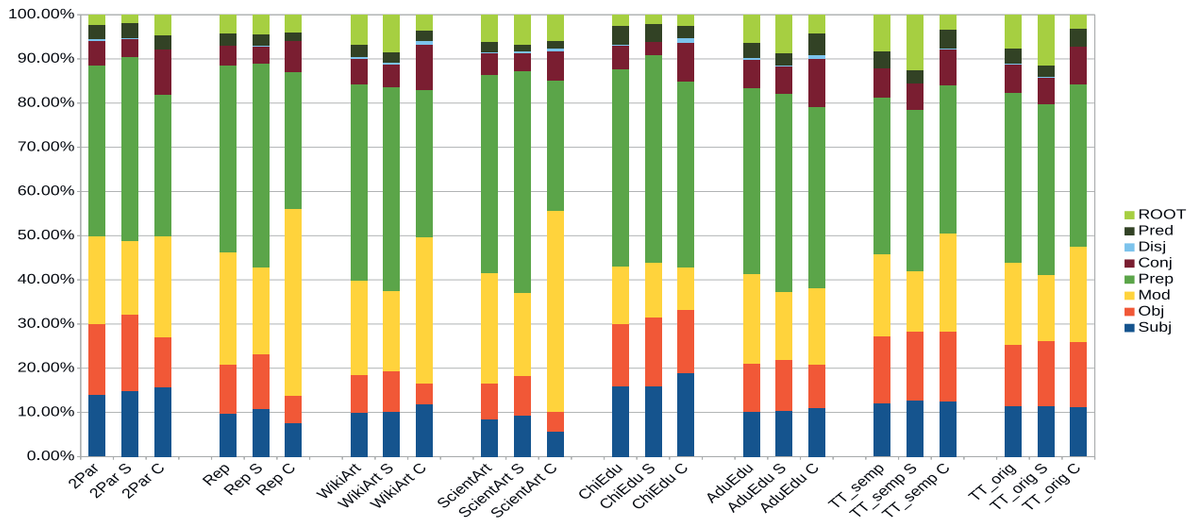


Figure 1: Distribution of typed syntactic dependencies linking nouns to their head across corpora. For each corpus, the first column refers to all nouns; the second one to the subset of *simple* nouns; the third one to the subset of *complex* nouns

	HeadDistance			**AvgChildren**			**AvgSibling**			**RootDistance**		
	All	Comp	Simp	All	Comp	Simp	All	Comp	Simp	All	Comp	Simp
2Par	2.252	2.342	2.256	1.318	1.218	1.345	1.675	1.956	1.580	2.969	2.816	2.993
Rep	2.210	2.271	2.272	1.213	0.979	1.323	1.558	1.509	1.564	4.197	4.314	4.131
Wiki	2.531	2.686	2.625	1.363	1.138	1.528	1.603	1.897	1.592	4.284	4.346	4.097
ArtScient	2.162	2.391	2.409	1.229	1.066	1.388	1.399	1.487	1.418	4.835	5.132	4.598
EduChi	2.177	2.338	2.171	1.311	1.303	1.353	1.523	1.621	1.458	3.408	3.387	3.388
EduAdu	2.598	2.875	2.695	1.440	1.375	1.560	1.654	1.715	1.640	4.269	4.483	4.143
TTsemp	2.167	2.334	2.172	1.342	1.335	1.470	1.690	1.789	1.659	3.017	2.953	2.882
TTorig	2.252	2.399	2.269	1.339	1.333	1.439	1.681	1.705	1.697	3.268	3.200	3.169

Table 1: Average value of the monitored syntactic features with respect to all nouns (All), to the subset of complex nouns (Comp) and to the subset of simple nouns (Simp) extracted from all the examined corpora.

role: 2.969; *Rep*: 4.197) and, to a lesser extent, the Educational one (*EduChi*: 3.408; *EduAdu*: 4.269). However, for the other monitored syntactic features, the *Wiki* corpus appears as slightly more difficult than its complex counterpart: it has nouns that are less close to their head (*Wiki*: 2.531; *ArtScient*: 2.162) and have a richer structure in terms of number of children (*Wiki*: 1.363; *ArtScient*: 1.229). With the exception of *root distance*, variations concerning other features within the Narrative genre are not statistically significant. This can be possibly due to the particular composition of the two selected corpora: indeed, both *Terence* and *Teacher* texts in their original version were already conceived for an audience of children and young students, and they were not greatly modified in their simplified version.

We finally assessed whether the variation of these features was statistically significant comparing the *simple* and the *complex* noun subset of the same corpus (Table 3). According to this dimension, we can observe that *complex* nouns have, on average, less dependents (*AvgChildren* feature) than *simple* ones, independently from the internal distinction within genre; on the contrary, they tend to occur more distant from the root, especially in the complex variety of Scientific prose (*ArtScient_Comp*: 5.132; *ArtScient_Simp*: 4.598).

5 Conclusion

While language complexity is a central topic in linguistic and computational linguistics research, it is typically addressed from a local perspective, where each subdomain is investigated in isola-

	HeadDistance			AvgChildren			AvgSibling			RootDistance		
	All	C	S	All	C	S	All	C	S	All	C	S
2Par vs Rep	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓	✓*	✓*	✓*
Wiki vs ArtScient	✓*	✓*	✓*	✓*	✗	✓*	✓*	✓*	✓*	✓*	✓*	✓*
EduChild vs EduAdu	✗	✗	✗	✓*	✗	✓	✓	✗	✗	✓*	✓*	✓*
TTsempl vs TTorig	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓*	✓	✓*
ArtScient vs EduAdu	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*
Rep vs ArtScient	✓*	✗	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*
Rep vs EduAdu	✓*	✓*	✓*	✓*	✓*	✓*	✓	✓	✗	✓	✗	✗
Rep vs TTorig	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*	✓*
TTorig vs ArtScient	✓*	✓*	✓*	✓*	✓*	✓	✓*	✓*	✓*	✓*	✓*	✓*
TTorig vs EduAdu	✗	✗	✗	✓*	✗	✓	✓*	✗	✓*	✓*	✓*	✓*
2Par vs EduChild	✓	✓*	✗	✓*	✓*	✗	✓*	✗	✓	✓*	✓*	✓*
2Par vs TTsemp	✗	✓*	✗	✓*	✓*	✗	✓	✗	✓*	✗	✗	✓*
2Par vs Wiki	✓*	✗	✓*	✓*	✓	✓*	✓*	✗	✓*	✓*	✓*	✓*
TTsemp vs EduChild	✗	✗	✗	✗	✗	✗	✓*	✗	✓*	✓*	✓*	✓*
TTsemp vs Wiki	✓*	✓*	✓*	✗	✓*	✗	✓*	✗	✓*	✓*	✓*	✓*
Wiki vs EduChild	✓*	✓*	✓*	✗	✓*	✓	✗	✗	✗	✓*	✓*	✓*

Table 2: Syntactic features that vary in a statistically significant way between the simple and the complex corpus of the same genre, between the complex corpora of each genre and between the simple corpora of each genre. "✗" means a non significant variation; "✓" means a significant variation at <0.05; "✓*" means a very significant variation at <0.01. All=all nouns; C=complex nouns; S=simple nouns.

	HeadDistance	AvgChildren	AvgSibling	RootDistance
2ParSostS vs 2ParSostC	✓*	✓*	✓*	✓*
RepSostS vs RepSostC	✓*	✓*	✗	✓*
WikiSostS vs WikiSostC	✗	✓*	✓*	✓*
ArtScientSostS vs ArtScientSostC	✓*	✓*	✗	✓*
EduChildSostS vs EduChildSostC	✗	✓	✗	✗
EduAduSostS vs EduAduSostC	✓	✓*	✗	✓*
TTsempSostS vs TTsempSostC	✓*	✗	✗	✗
TTorigSostS vs TTorigSostC	✓*	✓	✗	✗

Table 3: Linguistic features that vary in a statistically significant way between the *simple* and the *complex* nouns of the same corpus. "✗" means a non significant variation; "✓" means a significant variation at <0.05; "✓*" means a very significant variation at <0.01. All=all nouns; C=complex nouns; S=simple nouns.

tion. In this preliminary work, we have defined a method to study the interplay between lexical and syntactic complexity restricted to the nominal domain. We modeled the two notions in terms of frequency, with respect to lexical complexity, and of a set of parse tree features formalizing phenomena of syntactic complexity. Our approach was tested on corpora selected to be representative of different genres and different levels of complexity within each genre, in order to investigate whether noun complexity differently affects syntactic complexity according to the two dimensions. We observed e.g. that nouns tend to appear closer to the root in simple language varieties, independently from genre, while the effect of genre and linguistic complexity is less sharp with respect to the other considered features.

To have a deeper understanding of the observed tendencies we are currently carrying out a more in depth analysis focusing on fine-grained features of syntactic complexity, such as the depth of the nominal subtree. Further, we would like to enlarge this approach to test other constituents of the sentence, such as the verb.

Acknowledgments

The work presented in this paper was partially supported by the 2–year project (2017-2019) PERFORMA – Personalizzazione di pERcorsi FORMativi Avanzati, funded by Regione Toscana (Progetti Congiunti di Alta Formazione – POR FSE 2014-2020 Asse A – Occupazione) in collaboration with Meta srl company.

References

Giuseppe Attardi, Felice Dell'Orletta, Maria Simi, Joseph Turian. 2009. Accurate dependency parsing with a stacked multilayer perceptron. In *Proceedings of EVALITA 2009 - Evaluation of NLP and Speech Tools for Italian 2009*, Reggio Emilia, Italy, December 2009.

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

Dominique Brunato and Felice Dell'Orletta. 2017. On the order of words in Italian: a study on genre vs complexity. *International Conference on Dependency Linguistics (Depling 2017), 18-20 September 2017, Pisa, Italy.*

Cristina Burani. 2006. Morfologia: i processi. In: A. Laudanna and M. Voghera (cur.) *Il linguaggio. Strutture. Strutture linguistiche e processi cognitivi.* Bari, Laterza, 2006.

Isabella Chiari and Tullio De Mauro. 2014. The New Basic Vocabulary of Italian as a linguistic resource. *Proceedings of the First Italian Conference on Computational Linguistics (CLIC-IT)*, Pisa 15-19 dicembre 2014.

Andrea Cimino, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2013. Linguistic Profiling based on General–purpose Features and Native Language Identification. *Proceedings of Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, Atlanta, Georgia, June 13, pp. 207-215.

Kevyn Collins-Thompson. 2014. Computational Assessment of text readability. *Recent Advances in Automatic Readability Assessment and Text Simplification. Special issue of International Journal of Applied Linguistics*, 165:2, John Benjamins Publishing Company, 97-135.

Felice Dell'Orletta. 2009. Ensemble system for part-of-speech tagging. In *Proceedings of EVALITA 2009 - Evaluation of NLP and Speech Tools for Italian 2009*, Reggio Emilia, Italy, December 2009.

Holger Diessel. 2005. Competing motivations for the ordering of main and adverbial clauses. *Linguistics*, 43 (3): 449–470.

Richard Futrell, Kyle Mahowald and Edward Gibson. 2015. Quantifying word order freedom in dependency corpora. In *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, 91–100.

Edward Gibson. 1998. Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68:1–76.

Edward Gibson. 2000. The dependency Locality Theory: A distance–based theory of linguistic complexity. *Image, Language and Brain*, In W.O.A. Marants and Y. Miyashita (Eds.), Cambridge, MA: MIT Press, pp. 95–126.

Daniel Gildea and David Temperley. 2010. Do Grammars Minimize Dependency Length? *Cognitive Science*, 34(2):286310.

John A. Hawkins 1994. A performance theory of order and constituency. *Cambridge studies in Linguistics*, Cambridge University Press, 73.

Felice Dell'Orletta, Martjin Wieling, Andrea Cimino, Giulia Venturi, and Simonetta Montemagni. 2014. Assessing the Readability of Sentences: Which Corpora and Features. *Proceedings of the 9th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2014)*, Baltimore, Maryland, USA.

Matti Miestamo. 2008. Grammatical complexity in a crosslinguistic perspective. In: Miestamo M, Sinnemäki K. and Karlsson F. (eds), *Language Complexity: Typology, Contact Change*, Amsterdam: Benjamins, 23–41.

Randall James Ryder and Wayne H. Slater. 1988. The relationship between word frequency and word knowledge. *The Journal of Educational Research*, 81(5):312–317.

Kortmann Berndt and Szmrecsanyi Benedikt. 2012. *Linguistic Complexity. Second Language Acquisition, Indigenization, Contact*. Berlin, Boston: De Gruyter.

Yaqin Wang and Haitao Liu. 2017. The effects of genre on dependency distance and dependency direction. *Language Sciences*, 59, 135–157.

Chris Wild 1997. The Wilcoxon Rank-Sum Test. *University of Auckland, Department of Statistics*

Grammatical class effects in production of Italian inflected verbs

Maria De Martino, Azzurra Mancuso, Alessandro Laudanna
LaPSUS, Laboratory of Experimental Psychology, University of Salerno
Via Giovanni Paolo II, 132 Fisciano, SA, 84084, Italy
mdemartino@unisa.it amancuso@unisa.it alaudanna@unisa.it

Abstract

English. We report a picture-word interference (PWI) experiment conducted in Italian where target verbs were used to name pictures in presence of semantically related and unrelated distracters. The congruency of grammatical class between targets and distracters was manipulated and nouns and verbs were used as distracters. Consistently with previous studies, an expected semantic interference effect was observed but, interestingly, such an effect does not equally apply to target-distracter pairs sharing or not grammatical class information. This outcome seems to corroborate the hypothesis of the intervention of grammatical constraints in word production as explored in the PWI task.

Italiano. *Questo lavoro descrive un esperimento di interferenza figura-parola sull' italiano in cui le figure dovevano essere denominate usando verbi in presenza di distrattori semanticamente collegati o non collegati alla figura. È stata manipolata anche la congruenza di classe grammaticale tra target e distrattori; questi ultimi nella metà dei casi erano nomi e nell'altra verbi. In linea con studi precedenti, abbiamo ottenuto un effetto di interferenza semantica; il dato interessante è che quest'ultimo effetto interessa in modo differente le coppie target-distrattore congruenti o non congruenti per classe grammaticale. Questo risultato sembra corroborare l'ipotesi che nella di produzione di parole esplorata attraverso il compito di interferenza figura-parola giochino un ruolo le proprietà grammaticali delle parole.*

1. Introduction

Models of lexical access share the assumption that different kinds of linguistic information (semantic, orthographic-phonological, syntactic-grammatical, and so on) have different levels of lexical representation (Caramazza, 1997; Levelt, Roelofs and Meyer, 1999; Dell, 1986). The picture-word interference (PWI) paradigm has been widely exploited to test the dynamics of activation of different properties of words during lexical production. Such a task allows the observation of specific lexical effects by manipulating the linguistic relation between words to be used in a picture naming task and written distracter-words super-imposed to pictures. The basic assumption is that linguistic information of a distractor influences the time needed to select the appropriate word-form to name a picture. For instance, two well-known effects observed in PWI, the semantic interference and the phonological facilitation effects, are thought to reflect respectively the competition at the lexical level between the lexical representations of the target and the distracter and the co-activation of the phonemes shared by the target and the distracter during the phonetic encoding stage.

Scholars have also tried to investigate the activation of grammatical information in speech production through the PWI paradigm but conflicting evidence has been collected. For instance, Pechmann and Zerbst (2002), Pechmann and coll. (2004), Vigliocco and coll. (2005), Rodriguez-Ferreiro and coll. (2014), De Simone and Collina (2016) obtained grammatical class effects, while Mahon and coll. (2007), Iwasaki and coll. (2008) and Janssen and coll. (2010) did not. Arguably, the variability in the experimental evidence can be ascribed to heterogeneous methodologies across studies: for instance, results obtained by Vigliocco and coll. (2005) could be biased by their methodological choice to administer noun-distracters with determiners, while in the study of Rodriguez-Ferreiro and coll. (2014)

semantic categories (actions/objects/instruments) partially overlapped grammatical classes and a confound due to an imageability bias (Exp. 3) was present.
As a consequence, the intervention of grammatical constraints during production processes, as explored in PWI tasks, is still debated.
In this study on Italian we aimed at exploring the problem by trying to avoid possible confounds existing in previous studies.

2. Method

Participants: Thirty-six undergraduate students (28 females) from University of Salerno voluntarily took part in the experiment. They were all native speakers of Italian and they all had normal or corrected-to-normal vision. Their age ranged from 20 to 30 years (mean=22; sd=2.5). They served for a session lasting about 45 minutes.

Materials: Thirty-five black-and-white line drawings depicting actions were used as experimental items. Participants were instructed to name these pictures by using inflected verb forms (either present indicative, or 3rd singular person). These verbs constituted the target items. For each target-verb a semantically related distracter-verb and a semantically related distracter-noun were selected, so that a list of 35 distracter-verbs and a list of 35 distracter-nouns were built. The selected nouns and verbs were not affected by the semantic bias due to the object/action dichotomy. The semantic relatedness between targets and distracters was calculated on the basis of 2 measures: corpus-based automatic semantic metrics (WEISS, Word-embeddings Italian semantic spaces; Marelli, 2017) and subjective ratings on a 5 point Likert scale[1].

The same distracters were differently paired with the target verbs so that two lists of unrelated nominal (related-noun and unrelated-noun experimental conditions) and verbal (related-verb and unrelated-verb experimental conditions) distracters were created. Distracters in the four experimental conditions were matched for the main psycholinguistics variables: imageability, written form frequency (CoLFIS; Bertinetto et al., 2005) length, semantic relatedness. Formal orthographic or phonological overlap between targets and distracters was avoided. The mean values and standard deviations for each of these variables are reported in Table 1.

The experimental list was composed of 140 trials where the 35 target-verbs were accompanied by 70 verb-distracters (35 semantically related and 35 unrelated) and by 70 noun-distracters (35 semantically related and 35 unrelated). Two additional distracters were used as filler trials: for each target a related and an unrelated word were provided; these filler distracters differed from experimental distracters since they were word-class ambiguous items. Instances of all experimental conditions are reported in Table 2 and an example of experimental item is reported in Figure 1.

	Semantically related pairs		Semantically unrelated pairs	
	noun	verb	noun	verb
length	7.1 (1.6)	6.3 (1.4)	7.1 (1.6)	6.3 (1.4)
written form frequency	79.3 (92.3)	75.3 (97.7)	79.3 (92.3)	75.3 (97.7)
imageability	3.5 (0.6)	3.7 (0.6)	3.5 (0.6)	3.7 (0.6)
shared letters between targets and distracters	2 (1.1)	2 (1.1)	2 (1.1)	1.6 (1.0)
subjective semantic relatedness ratings	3.3 (0.9)	3.5 (1.03)	1.4 (0.4)	1.4 (0.4)
WEISS metrics	0.7 (0.1)	0.6 (0.2)	0.9 (0.1)	0.9 (0.1)

Table 1: Mean values and standard deviations (in parenthesis) of distracters' characteristics

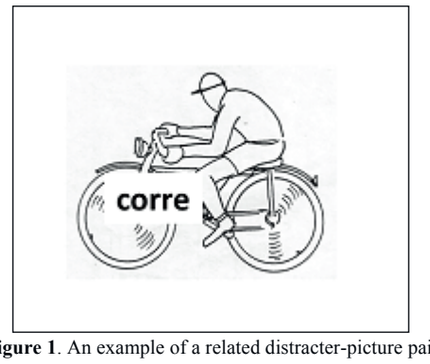


Figure 1. An example of a related distracter-picture pair

[1] The first measure provided objective values, based on distributional estimates, for the semantic distance between each target-word and its distracter. The second measure allowed us to ascertain to what extent the specific word sense evoked by the picture was related to the distracter-word.

Distracters	Related noun:	*frittura* (frying)
	Related verb:	*frigge* (he/she fries)
	Unrelated noun:	*rumore* (noise)
	Unrelated verb:	*sente* (he/she listens to)
Target	*cuoce* (he/she cooks)	

Table 2. Distracter-target pairs

In order to prevent any strategic bias due to semantic and/or grammatical relationships among targets and distracters, 15 additional pictures were used as filler targets and were presented with 6 different distracters. The whole list of both experimental and filler target-distracter pairs was composed of 300 trials: 33% were semantically related trials and 67% were unrelated trials.

Procedure: The participants were tested individually; an experimental session consisted of three parts: a familiarization, a practice and an experimental phase. The E-Prime software 2.0 (Psychology Software Tools, Inc., Pittsburgh, PA) was used.

At the beginning of the experiment, each participant was familiarized with the whole set of experimental and filler pictures in an untimed picture naming session. In this phase, the pictures were presented on the computer screen with a superimposed row of Xs to simulate the distracter word. Participants learned to produce the targets upon presentation of the corresponding pictures. If participants named a picture with a verb that differed from the one designed as the target by experimenters, a feedback was given: the expected verb was provided to participants and they were invited to use it in the experimental session.

Following the familiarization phase, a practice block was administered where participants were asked to name each picture as inflected verb forms (present indicative 3rd singular person, e.g. *beve*, he/she drinks) and were instructed to respond as quickly and accurately as possible, while ignoring the distracter word. The experimenter was seated behind the participant and recorded errors and equipment failures. The stimuli presented in the training phase were part of the filler set.

The stimuli appeared on a video display unit controlled by a personal computer. Reaction times from the appearance of the stimuli to the onset of articulation were collected by a voice key connected to the computer and participant responses were recorded. Upon a response, the picture and the distracter disappeared from the screen. Both the presentation of the stimuli and the recording of the responses were managed by the E-Prime software 2.0. The responses of the participants were checked for accuracy by an experimenter.

Each single trial consisted of the following events: a fixation cross presented at the center of the screen for 300 ms; the stimulus until the response or for a maximum of 2.5 seconds; a feedback mask signaling the activation of the voice key of 500ms, a blank interval of 500 ms. The SOA between pictures and distracter-words was 0 ms.

Words pronounced incorrectly, non-expected picture names, hesitations in giving the responses, word fragments, omissions, verbal dysfluencies and responses given after the deadline were scored as errors. Invalid responses (e.g., trials in which the voice key was triggered by external noise) and responses shorter than 400 ms were considered as missing data.

At the end of the practice phase, the experiment started and 6 experimental blocks of 50 trials (35 experimental items and 15 filler items) were presented, for a total of 300 trials. An equal number of items from each experimental condition was included in every block. Blocks were counterbalanced across participants. In each block, stimuli underwent a randomization governed by the E-Prime software 2.0.

3. Results

An analysis of variance (ANOVA) was performed on naming latencies and accuracy rates by subjects (F1) and by items (F2) with the distractor type (four levels) as a variable. For the sake of conciseness only the statistically significant analyses will be reported and discussed.

A main effect of semantic relatedness has been observed both in the ANOVA by participants ($F1(1, 35) = 4.56$, $p < .05$) and by items ($F2(1, 30) = 4.46$, $p < .05$) on response latencies. Responses to target verbs were slower when they were accompanied by semantically related distracters (+17 ms).

Neither effects of grammatical class nor interaction between grammatical class and semantic relation were found.

Two-tailed t tests comparing the semantic interference effect within the grammatical class congruent and non-congruent target/distracter pairs revealed that the semantic interference effect reaches the statistical significance with noun-distracters (+24 ms, p = .02) but not with verb-distracters (+9 ms, p = .43). The results are graphically shown in Table 3.

	Noun distracters	Verb distracters
Related	1020 ms (125)	1011 ms (121)
Unrelated	996 ms (107)	1002 ms (111)

Table 3. Mean response latencies and standard deviations (in parenthesis) for all conditions

4. Conclusions

One of the aim of the present experiment was to overcome some limitations of previous investigations. The following constraints were adopted:

1. We contrasted the production of verbs when presented with semantically related and unrelated distracters: the expected semantic interference effect guaranteed for the reliability of the paradigm.
2. We selected experimental materials where the differences between grammatical classes in terms of their semantic domain (objects (nouns) vs. actions (verbs)) was kept under control.
3. Word-class ambiguous items were excluded by experimental materials.
4. Inflected finite verbal-forms were used both as targets and distracters: these verbal forms allow to maximize the difference between nouns and verbs[2]. Actually, the Italian inflected form "*amavo*" (indicative, imperfect, 1st singular person, I used to love), is composed of a stem, "*am*-", which conveys the core meaning of the verb, the vowel "-*a*-", which specifies the inflectional pattern compatible with the verbal stem, the segment "–*v*-", which encodes mood and tense information, and the segment "-*o*" which encodes person and number information. None of these features, with the exception of meaning and number features, can be part of the lexical representation of noun-forms. This latter manipulation has relevant consequences on the detection of grammatical class effect in PWI, since it has been demonstrated that, when finite verbs have to be produced, the naming context sets the response-relevant criterion on the grammatical class of verbs and then noun-distracters tend to interfere significantly more than verb-distracters (De Martino & Laudanna, 2017)[3].

Consistently with previous PWI evidence, our experiment replicated a reliable semantic interference effect. This finding confirms that the selection of an oral target response is slowed-down by the activation of a semantically-related distracter because the lexical system has to manage the level of activation of target lexical competitors, including the highly activated semantically related distracter word. Interestingly, we observed that, at least when pictures have to be named by using inflected verb forms, such an effect does not equally affect all semantically related target-distracter pairs: related pairs sharing grammatical class information do not exhibit significant semantic interference but grammatical-class incongruent pairs do.

In conclusion, our data suggest that the PWI task is sensitive to the manipulation of grammatical class information. In other words, such a pattern of results is compatible with the intervention of grammatical constraints during production processes, as explored in the PWI task.

References

Bertinetto, P. M., Burani, C., Laudanna, A., Marconi, L., Ratti, D., Rolando, C., & Thornton, A. M. (2005). *Corpus e Lessico di Frequenza*

[2] The distinction between finite and non-finite moods is motivated on morphological and syntactic grounds: finite-forms are inflected for person and in syntactic context they are used as verbal predicates. Conversely, non-finite forms lack for person inflection and are used in periphrastic construction or in combination with auxiliary verbs to assemble the "composed tenses" of the paradigm. Under certain circumstances, non-finite forms undergo syntactic transcategorization and behave as nouns or adjectives: "*mi piace ballare* [infinitive]", (I love dancing). "*I partecipanti* [present participle], *sono pronti*" (participants are ready); "*tre gare vinte* [past participle, from "*vincere*"] *e cinque perse* [past participle, from "*perdere*"], (three competitions won and five lost).

[3] This result was obtained regardless of semantic relation between targets and distracters.

dell'Italiano Scritto (CoLFIS). http://linguistica.sns.it/CoLFIS/Home.htm

Caramazza, A. (1997). How many levels of processing are there in lexical access?.*Cognitive Neuropsychology*, *14*(1), 177-208.

Dell, G. S. (1986). A spreading-activation theory of retrieval in sentence production. *Psychological Review*, *93*(3), 283.

De Martino, M., & Laudanna, A. (2017). The role of grammatical properties in the word-picture interference paradigm: data from single verbs production in Italian. In: *Abstracts of the 20th Conference of the European Society for Cognitive Psychology* (198-198).

De Simone, F., & Collina, S. (2016). The Picture–Word Interference Paradigm: Grammatical Class Effects in Lexical Production. *Journal of Psycholinguistic Research*, *45*(5), 1003-1019.

Iwasaki, N., Vinson, D. P., Vigliocco, G., Watanabe, M., & Arciuli, J. (2008). Naming action in Japanese: Effects of semantic similarity and grammatical class. *Language and Cognitive Processes*, *23*(6), 889-930.

Janssen, N., Melinger, A., Mahon, B. Z., Finkbeiner, M., & Caramazza, A. (2010). The word class effect in the picture–word interference paradigm. *The Quarterly Journal of Experimental Psychology*, *63*(6), 1233-1246.

Levelt, W. J., Roelofs, A., & Meyer, A. S. (1999). A theory of lexical access in speech production. *Behavioral and Brain Sciences*, *22*(1), 1-38.

Mahon, B. Z., Costa, A., Peterson, R., Vargas, K. A., & Caramazza, A. (2007). Lexical selection is not by competition: a reinterpretation of semantic interference and facilitation effects in the picture-word interference paradigm. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 33 (3), 503-535.

Marelli, M. (2017). Word-Embeddings Italian Semantic Spaces: A semantic model for psycholinguistic research. *Psihologija*, *50*(4), 503-520.

Pechmann, T., & Zerbst, D. (2002). The activation of word class information during speech production. *Journal of Experimental Psychology: Learning, Memory, and Cognition,* 28 (1), 233–243.

Pechmann, T., Garrett, M., & Zerbst, D. (2004). The time course of recovery for grammatical category information during lexical processing for syntactic construction. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *30*(3), 723.

Rodríguez-Ferreiro, J., Davies, R., & Cuetos, F. (2014). Semantic domain and grammatical class effects in the picture–word interference paradigm. *Language, Cognition and Neuroscience*, *29*(1), 125-135.

Vigliocco, G., Vinson, D. P., & Siri, S. (2005). Semantic similarity and grammatical class in naming actions. *Cognition*, *94*(3), B91-B100.

Integrating Terminology Extraction and Word Embedding for Unsupervised Aspect Based Sentiment Analysis

Luca Dini
Innoradiant
Grenoble, France
luca.dini@innoradiant.com

Paolo Curtoni
Innoradiant
Grenoble, France
paolo.curtoni@innoradiant.com

Elena Melnikova
Innoradiant
Grenoble, France
elena.melnikova@innoradiant.com

Abstract

English. In this paper we explore the advantages that unsupervised terminology extraction can bring to unsupervised Aspect Based Sentiment Analysis methods based on word embedding expansion techniques. We prove that the gain in terms of F-measure is in the order of 3%.

Italiano. *Nel presente articolo analizziamo l'interazione tra syistemi di estrazione "classica" terminologica e systemi basati su techniche di "word embedding" nel contesto dell'analisi delle opinioni. Domostreremo che l'integrazione di terminogie porta un guadagno in F-measure pari al 3% sul dataset francese di Semeval 2016.*

1 Introduction

The goal of this paper is to bring a contribution on the advantage of exploiting terminology extraction systems coupled with word embedding techniques. The experimentation is based on the corpus of Semeval 2016. In a previous work, summarized in section 4, we reported the results of a system for Aspect Based Sentiment Analysis (ABSA) based on the assumption that in real applications a domain dependent gold standard is systematically absent. We showed that by adopting domain dependent word embedding techniques a reasonable level of quality (i.e. acceptable for a proof of concept) in terms of entity detection could be achieved by providing two seed words for each targeted entity. In this paper we explore the hypothesis that unsupervised terminology extraction approaches could further improve the quality of the results in entity extraction.

The paper is organized as follows: In section 2 we enumerate the goal of the research and the industrial background justifying it. In section 3 we provide a state of the art of ABSA particularly focused towards unsupervised ABSA and its relationship to terminology extraction. In section 4 we summarize our previous approach in order to provide a context for our experimentation. In section 5 we prove the benefit of the integration of unsupervised terminology extraction with ABSA, whereas in 6 we provide hints for further investigation.

2 Background

ABSA is a task which is central to a number of industrial applications, ranging from e-reputation, crisis management, customer satisfaction assessment etc. Here we focus on a specific and novel application, i.e. capturing the voice of the customer in new product development (NPD). It is a well-known fact that the high rate of failure (76%, according to Nielsen France, 2014) in launching new products on the market is due to a low consideration of perspective users' needs and desires. In order to account for this deficiency a number of methods have been proposed ranging from traditional methods such as KANO (Wittel et al., 2013) to recent lean based NPD strategies (Olsen, 2015). All are invariantly based on the idea of collecting user needs with tools such as questionnaire, interviews and focus groups. However with the development of social networks, reviews sites, forums, blogs etc. there is another important source for capturing user insights for NPD: users of products (in a wide sense) are indeed talking about them, about the way they use them, about the emotions they raise. Here it is where ABSA becomes central: whereas for applications such as e-reputation or brand monitoring capturing just the sentiment is largely enough for the specific purpose, for NPD it is crucial to capture the entity an opinion is referring to and the specific feature under judgment.

ABSA for NPD is a novel technique and as such it might trigger doubts on its adoption: given the investments on NPD (198 000 M€ only in the cosmetics sector) it is normal to find a certain reluctance in abandoning traditional methodologies for voice of the customer

collection in favor of social network based AB-ABSA. In order to contrast this reluctance, two conditions need to be satisfied. On the one hand, one must prove that ABSA is feasible and effective in a specific domain (Proof of Concept, POC); on the other hand the costs of a high quality in-production system must be affordable and comparable with traditional methodologies (according to Eurostat the spending of European PME in the manufacturing sector for NPD will be about 350,005.00 M€ in 2020, and PME usually have limited budget in terms of "voice of the customer" spending).

If we consider the fact that the range of product/services which are possible objects of ABSA studies is immense[1], it is clear that we must rely on almost completely unsupervised technologies for ABSA, which translates in the capability of performing the task *without* a learning corpus.

3 State of the Art

3.1 Semeval2016's overview

SemEval is " an ongoing series of evaluations of computational semantic analysis systems" [2], organized since 1998. Its purpose is to evaluate semantic analysis systems. ABSA (Aspect Based Sentiment Analysis) was one of the tasks of this event introduced in 2014. This type of analysis provides information about consumer opinions on products and services which can help companies to evaluate the satisfaction and improve their business strategies. A generic ABSA task consists to analyze a corpus of unstructured texts and to extract fine-grained information from the user reviews. The goal of the ABSA task within SemEval is to directly compare different datasets, approaches and methods to extract such information (Pontiki et al., 2016).

In 2016, ABSA provided 39 training and testing datasets for 8 languages and 7 domains. Most datasets come from customer reviews (especially for the domains of restaurants, laptops, mobile phones, digital camera, hotels and museums), only one dataset (telecommunication domain) comes from tweets. The subtasks of the sentence-level ABSA, were intended to identify all the opinion tuples encoding three types of information: Aspect category, Opinion Target Expression (OTE) and Sentiment polarity. Aspect is in turn a pair (E#A) composed of an Entity and an Attribute. Entity and attributes, chosen from a special inventory of entity types (e.g. "restaurant", "food", etc.) and attribute labels (e.g. "general", "prices", etc.) are the pairs towards which an opinion is expressed in a given sentence. Each E#A can be referred to a linguistic expression (OTE) and be assigned one polarity label.

The evaluation assesses whether a system correctly identifies the aspect categories towards which an opinion is expressed. The categories returned by a system are compared to the corresponding gold annotations and evaluated according to different measures (precision (P), recall (R) and F-1 scores). System performance for all slots is compared to baseline score. Baseline System selects categories and polarity values using Support Vector Machine (SVM) based on bag-of-words features (Apidianaki et al., 2016).

3.2 Related works on unsupervised ABSA

Unsupervised ABSA. Traditionally, in ABSA context, one problematic aspect is represented by the fact that, given the non-negligible effort of annotation, learning corpora are not as large as needed, especially for languages other than English. This fact, as well as extension to "unseen" domains, pushed some researchers to explore unsupervised methods. Giannakopoulos et al. (2017) explore new architectures that can be used as feature extractors and classifiers for Aspect terms unsupervised detection.

Such unsupervised systems can be based on syntactic rules for automatic aspect terms detection (Hercig et al., 2106), or graph representations (García-Pablos et al., 2017) of interactions between aspect terms and opinions, but the vast majority exploits resources derived from distributional semantic principles (concretely, word embedding).

The benefits of word embedding used for ABSA were successfully shown in (Xenos et al., 2016). This approach, which is nevertheless supervised, characterizes an unconstrained system (in the Semeval jargon a system accessing information not included in the training set) for detecting Aspect Category, Opinion Target expression and Polarity. The used vectors were produced using the skip-gram model with 200 dimensions and were based on multiple ensembles, one for each E#A combination. Each ensemble returns the combinations of the scores of constrained and unconstrained systems. For Opinion Target

[1] The site of UNSPC reports more than 40,000 categories of products (https://www.unspsc.org).

[2] https://aclweb.org/aclwiki/SemEval_Portal, seen on 05/24/2018

expression, word embedding based features extend the constrained system. The resulting scores reveal, in general, rather high rating position of the unconstrained system based on word embedding. Concerning the advantages derived from the use of pre-trained in domain vectors, they are also described in (Kim, 2014), who makes use of convolutional neural networks trained on top of pre-trained word vectors and shows good performances for sentence-level tasks, and especially for sentiment analysis

Some other systems represent a compromise between supervised and unsupervised ABSA, i.e. semi-supervised ABSA systems, such an almost unsupervised system based on topic modelling and W2V (Hercig et al., 2016), and W2VLDA (García-Pablos et al., 2017). The former uses human annotated datasets for training, but enrich the feature space by exploiting large unlabeled corpora. The latter combines different unsupervised approaches, like word embedding and Latent Dirichlet Allocation (LDA, Blei et al., 2003) to classify the aspect terms into three Semeval categories. The only supervision required by the user is a single seed word per desired aspect and polarity. Because of that, the system can be applied to datasets of different languages and domains with almost no adaptation.

Relationship with Term Extraction. Automatic Terminology Extraction (ATE) is an important task in NLP, because it provides a clear footprint of domain-related information. All ATE methods can be classified into linguistic, statistical and hybrid (Cabré-Castellvi et al., 2001).

The relationship between word embedding and ATE method is successfully explored for tasks of term disambiguation in technical specification documents (Merdy et al., 2016). The distributional neighbors of the 16 seed words were evaluated on the basis of the three corpora of different size: small (200,000 words), medium (2 M words) and large (more than 200 M words). The results of this study show that the identification of generic terms is more relevant in the large sized corpora, since the phenomenon is very widespread over the contexts. For specified terms, medium and large sized corpora are complementary. The specialized medium corpora brings a gain value by guaranteeing the most relevant terms. As for the small corpora, it does not seem to give usable results, whatever the term. Thus, the authors conclude that word2vec is an ideal technique to constitute semi-automatically term lexicon from very large corpora, without being limited to a domain.

Word2vec's methods (such as skip-gram and CBOW) are also used to improve the extraction of terms and their identification. This is done by the composed filtering of Local-global vectors (Amjadian et al., 2016). The global vectors were trained on the general corpus with GloVe (Pennington et al., 2014), and the local vectors on the specific corpus with CBOW and Skip-gram. This filter has been made to preserve both specific-domain and general-domain information that the words may contain. This filter greatly improves the output of ATE tools for a unigram term extraction.

The W2V method seems useful for the task of categorizing terms using the concepts of an ontology (Ferré, 2017). The terms (from medical texts) were first annotated. For each term an initial vector was generated. These term vectors, embedded into the ontology vector space, were compared with the ontology concept vectors. The calculated closest distance determines the ontological labeling of the terms.

Word2vec method is used also to emulate a simple ontology learning system to execute term and taxonomy extraction from text (Wohlgenannt and Minic, 2016). The researchers apply the built-in word2vec similarity function to get terms related to the seed terms. But the minus-side of the results shows that the candidates suggested by word2vec are too similar terms, as plural forms or near synonyms. On the other hand, the evaluation of word2vec for taxonomy building gave the accuracy of around 50% on taxonomic relation suggestion. Being not very impressive, the system will be improved by parameter settings and bigger corpora.

In the experiments described in this paper we exploit only the Skip-gram approach based on the word2vec implementation. It is important to notice that this choice is not due to a principled decision but to not functional constraints related the fact that that algorithm has a java implementation, is reasonably fast and it is already integrated with Innoradiant NLP pipeline.

4 Previous Investigations

The experiments described in Dini et al. (under review), have been performed by using Innoradiant's Architecture for Language Analytics (henceforth IALA). The platform implements a standard pipelined architecture composed of classical NLP modules: Sentence Splitting → Tokenization → POS tagging → lexicon access → Dependency Parsing → Feature identification

→ Attitude analysis. Inspired by Dini et al. (2017) and Dini and Bittar (2016), sentiment/attitude analysis in IALA is mainly symbolic. The basic idea is that dependency representations are an optimal input for rules computing sentiments. The rule language formalism is inspired by Valenzuela-Escárcega et al. (2015) and thanks to its template filling capability, in several cases, the grammar is able to identify the **perceiver** of a sentiment and, most importantly the **cause**, of the sentiment, represented by a word in an appropriate syntactic dependency with the sentiment-bearing lexical item. For instance the representation of the opinion in *I hate black coffee.* would be something such as:

```
<Opinion trigger="1" perceiver="0" cause="3">.
```

(where integers represent position of words in a CONLL like structure).

By default entities (which are normally products and services under analysis) are identified since early processing phases by means of regular expressions. This choice is rooted in the fact that by acting at this level multiword entities (such as *hydrating cream*) are captured as single words since early stages.

The goal of the Dini et al. (2018) work was to minimize the domain configuration overhead by i) expanding automatically the polarity lexicon to increase polarity recall and ii) to perform entity recognition by providing only two words (seeds) for each target entity.

Both goals were achieved by exploiting a much larger corpus than Semeval, obtained by automatically scraping restaurant review from TripAdvisor. The final corpus was composed of 3,834,240 sentence and 65,088,072 lemmas. From this corpus we obtain a word2vec resource by using the DL4j library (skip-gram). The resource (W2VR, henceforth) was obtained by using lemma rather than surface forms. Relevant training parameters for reproducing the model are described in that paper.

We skip here the description of i) (polarity expansion) as in the context of the present work *we kept polarity exactly as it was* in Dini & al. (2018)[3]. We just mention the achieved results on polarity only detection which were a precision of 0.78185594 and a recall of 0.54541063 (F-measure: 0.6425726). These numbers are important because in our approach a positive match is always given by a positive match of polarity *and* a correct entity identification (in other words a perfect entity detection system could achieve a maximum of 0.64 precision).

[3] Some previous works on unsupervised polarity lexicon acquisition for sentiment analysis were done in (Castellucci et al., 2016; Basili et al., 2017)

4.1 Entity Matching

Entity matching was achieved by manually associating two seed words to each Semeval entity (RESTAURANT, FOOD, DRINK, etc.) and then applying the following algorithm:

- Associate each entity to the average vector of the seed words (**e-vect.** E.g. evect(FOOD)=avg(vect(*cuisine*),vect(*pizza*)).
- If a syntactic cause is found by the grammar (as in "I liked the **meal**")assign it the entity associated to the closest e-vect.
- Otherwise compute the average vector of n words surrounding the opinion trigger and assign the entity associated to the closest e-vect.

With n=35 we obtain precision= 0.47914252, recall= 0.4888 and F-measure=0.3998.

5 Integrating terminology

A possible path to improve results in entity assignment can be found in the usage of "synonyms" in the computation of the set of e-vect. These can again be obtained from W2VR by selecting the *n* closest world to the average of the seeds and using them in the computation of the e-vect. Expectedly, the value of *n* can influence the result as shown in Figure **1**.

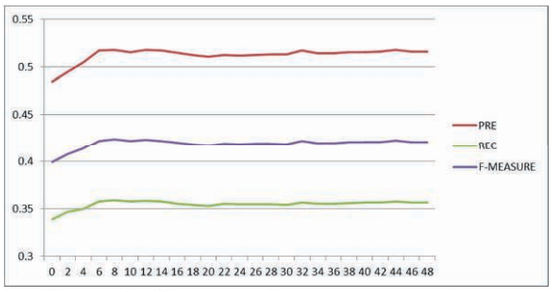


Figure 1. Changes of measures accoring to different top N closest worlds (without terminology).

We notice that best results are achieved by using a set of closest world around 10: after that threshold the noise caused by "false synonyms" or associated common words causes a decay in

the results. We also notice that overall the results are better than the original seed-only method, as now we obtain precision: 0.51 recall: 0.35 F-measure: 0.42. Here the positive fact is not only a global raise of the f-measure, but the fact that this is mainly caused by an increased precision, which according to Dini et al. (2018) is *the* crucial point in POC level applications.

As a way to remedy to the noise caused by an unselective use of the *n* closest words coming from W2VR we decide to explore an approach that filters them according to the words appearing as *terms* in a terminology obtained from unsupervised terminology extraction system. To this purpose we adopted the software TermSuite (Cram & Daille, 2016) which implements a classic two steps model of identification of term candidates and their ranking. In particular TermSuite is based on two main components, a UIMA Tokens Regex for defining terms and variant patterns over word annotations, and a grouping component for clustering terms and variants that works both at morphological and syntactic levels (for more details cf. Cram & Daille, 2016). The interest of using this resource for filtering results from W2VR is that "quality word" lists are obtained with the adoption of methods fundamentally different from W2V approach and heavily based on language dependent syntactic patterns.

We performed the same experiments as W2VR expansion for the computation of e-vect, with the only difference that now the top *n* must appear as closest terms in W2VR *and* as terms in the terminology (The W2VR parameters, including corpus are described in section 4; the terminology was obtained from the same corpus about restaurants). The results are detailed in Figure **2**.

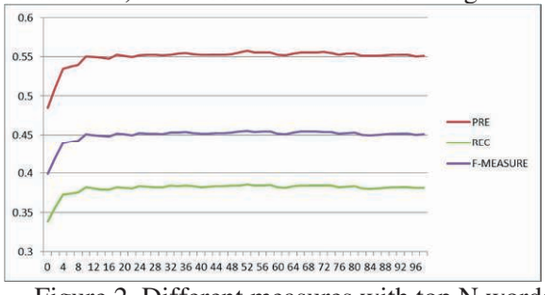


Figure 2. Different measures with top N words filtered with terminology.

We notice that all scores increase significantly. In particular at top *n*=10 we obtain P=0.550233483, R=0.381750288 and F=0.450762752, which represents a 5% increase (in F-measure) w.r.t. the results presented in Dini et al. (2018).

6 Conclusions

Many improvements can be conceived to the method presented here, especially concerning the computation of the vector associated to the opinionated windows, both in terms of size, directionality and consideration of finer grained features (e.g. indicators of a switch of topic). However our future investigation will rather be oriented towards full-fledged ABSA, i.e. taking into account not only Entities, but also Attributes. Indeed, if we consider that the 45% F measure is obtained on a corpus where only 66% sentences were correctly classified according to the sentiment and if we put ourselves in a Semeval perspective where entity evaluation is provided with respect to a "gold sentiment standard" we achieve a F-score of 68%, which is fully acceptable for an almost unsupervised system.

References

Ehsan Amjadian, Diana Inkpen, T.Sima Paribakht and Farahnaz Faez. 2016. Local-Global Vectors to Improve Unigram Terminology Extraction. *Proceedings of the 5th International Workshop on Computational Terminology,* Osaka, Japan, Dec 12, 2016, 2-11.

Marianna Apidianaki, Xavier Tannier and Cécile Richart. 2016. Datasets for Aspect-Based Sentiment Analysis in French. *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. Portorož, Slovenia.

Roberto Basili, Danilo Croce and Giuseppe Castellucci. 2017. Dynamic polarity lexicon acquisition for advanced Social Media analytics. *International Journal of Engineering Business Management*, Volume 9, 1-18.

David M. Blei, Andrew Y. Ng and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *The Journal of machine Learning research*, Volume 3, 993–1022.

M. Teresa Cabré Castellví, Rosa Estopà Bagot, Jordi Vivaldi Palatresi. 2001. Automatic term detection: a review of current systems. In Bourigault, D. Jacquemin, C. L'Homme, M-C. 2001. *Recent Advances in Computational Terminology*, 53-88.

Giuseppe Castellucci, Danilo Croce and Roberto Basili. 2016. A Language Independent Method for Generating Large Scale Polarity Lexicons. *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC'16)*, Portoroz, Slovenia, 38-45.

Damien Cram and Béatrice Daille. 2016. TermSuite: Terminology Extraction with Term Variant Detection. *Proceedings of the 54th Annual Meeting of*

the Association for Computational Linguistics—System Demonstrations, Berlin, Germany, 13–18.

Luca Dini, Paolo Curtoni and Elena Melnikova. 2018. Portability of Aspect Based Sentiment Analysis: Thirty Minutes for a Proof of Concept. Submitted to: *The 5th IEEE International Conference on Data Science and Advanced Analytics*. DSAA 2018, Turin.

Luca Dini, André Bittar, Cécile Robin, Frédérique Segond and M. Montaner. 2017. SOMA: The Smart Social Customer Relationship Management. *Sentiment Analysis in Social networks. Chapter 13*. 197-209. DOI: 10.1016/B978-0-12-804412-4.00013-9.

Luca Dini and André Bittar. 2016. Emotion Analysis on Twitter: The Hidden Challenge. *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016*, Portorož, Slovenia, 2016.

Arnaud Ferré. 2017. Représentation de termes complexes dans un espace vectoriel relié à une ontologie pour une tâche de catégorisation. *Rencontres des Jeunes Chercheurs en Intelligence Artifcielle (RJCIA 2017)*, Jul 2017, Caen, France.

Aitor García-Pablos, Montse Cuadros and German Rigau. 2017. W2VLDA: Almost unsupervised system for Aspect Based Sentiment Analysis. *Expert Systems with Applications, (91):* 127-137. arXiv:1705.07687v2 [cs.CL], 18 jul 2017.

Athanasios Giannakopoulos, Diego Antognini, Claudiu Musat, Andreea Hossmann and Michael Baeriswyl. 2017. Dataset Construction via Attention for Aspect Term Extraction with Distant Supervision. *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*

Tomáš Hercig, Tomáš Brychcín, Lukáš Svoboda, Michal Konkol and Josef Steinberger. 2016. Unsupervised Methods to Improve Aspect-Based Sentiment Analysis in Czech. *Computación y Sistemas, vol. 20*, No. 3, 365-375.

David Jurgens and Keith Stevens. 2010. The S-Space package: An open source package for word space models. *Proceedings of the ACL 2010 Systel Demonstrations*, 30-35.

Noriaki Kano, Nobuhiku Seraku, Fumio Takahashi, Shinichi Tsuji, 1984. Attractive Quality and Must-Be Quality. *Hinshitsu*: *The Journal of the Japanese Society for Quality Control*, *14(2) :* 39-48.

Emilie Merdy, Juyeon Kang and Ludovic Tanguy. 2016. Identification de termes flous et génériques dans la documentation technique : expérimentation avec l'analyse distributionnelle automatique. *Actes de l'atelier "Risque et TAL" dans le cadre de la conférence TALN.*

Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*

Tomas Mikolov, Wen-tau Yih and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. *Proceedings of NAACL-HLT 2013*, 746–751.

Dan R. Olsen. 2015. *The Lean Product Playbook: How to Innovate with Minimum Viable Products and Rapid Customer Feedback*. John Wiley & Sons Inc: New York, United States.

Jeffrey Pennington, Richard Socher and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532-1543.

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clecq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeny Kotelnikov, Nuria Bel, Salud M. Jiménez-Zafra, Gülşen Eryiğit. 2016. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*. San Diego, USA.

Marco A. Valenzuela-Escárcega, Gustave V. Hahn-Powell and Mihai Surdeanu. 2015. Description of the Odin Event Extraction Framework and Rule Language. arXiv:1509.07513v1 [cs.CL], 24 Sep 2015, version 1.0, 2015.

Lars Witell, Martin Löfgren and Jens J. Dahlgaard. 2013. Theory of attractive quality and the Kano methodology – the past, the present, and the future. *Total Quality Management & Business Excellence*, (24), 11-12:1241-1252.

Gerhard Wohlgenannt, Filip Minic. 2016. Using word2vec to Build a Simple Ontology Learning System. *Proceedings of the ISWC 2016 co-located with 15th International Semantic Web Conference (ISWC 2016)*. Vol-1690. Kobe, Japan, October 19, 2016

Dionysios Xenos, Panagiotis Theodorakakos, John Pavlopoulos, Prodromos Malakasiotis, Ion Androutsopoulos. 2016. AUEB-ABSA at SemEval-2016 Task 5: Ensembles of Classifiers and Embeddings for Aspect Based Sentiment Analysis. *Proceedings of SemEval-2016*, San Diego, California, 312–317.

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP),* 1746–1751. arXiv:1408.5882

A Linguistic Failure Analysis of Classification of Medical Publications: A Study on Stemming vs Lemmatization

Giorgio Maria Di Nunzio
Dept. of Information Engineering
University of Padua, Italy
dinunzio@dei.unipd.it

Federica Vezzani
Dept. of Languages and Literary Studies
University of Padua, Italy
federica.vezzani@phd.unipd.it

Abstract

English. Technology-Assisted Review (TAR) systems are essential to minimize the effort of the user during the search and retrieval of relevant documents for a specific information need. In this paper, we present a failure analysis based on terminological and linguistic aspects of a TAR system for systematic medical reviews. In particular, we analyze the results of the worst performing topics in terms of recall using the dataset of the CLEF 2017 eHealth task on TAR in Empirical Medicine.

Italiano. I sistemi TAR (Technology-Assisted Review) sono fondamentali per ridurre al minimo lo sforzo dell'utente che intende ricercare e recuperare i documenti rilevanti per uno specifico bisogno informativo. In questo articolo, presentiamo una *failure analysis* basata su aspetti terminologici e linguistici di un sistema TAR per le revisioni sistematiche in campo medico. In particolare, analizziamo i topic per i quali abbiamo ottenuto dei risultati peggiori in termini di recall utilizzando il dataset di *CLEF 2017 eHealth task on TAR in Empirical Medicine*.

1 Introduction

The Cross Language Evaluation Forum (CLEF) (Goeuriot et al., 2017) Lab on eHealth has proposed a task on Technology-Assisted Review (TAR) in Empirical Medicine since 2017. This task focuses on the problem of systematic reviews in the medical domain, that is the retrieval of all the documents presenting some evidence regarding a certain medical topic. This kind of problem is also known as total recall (or total sensitivity) problem since the main goal of the search is to find possibly all the relevant documents for a specific topic.

In this paper, we present a failure analysis based on terminological and linguistic aspects of the system presented by (Di Nunzio, 2018) on the CLEF 2017 TAR dataset. This system uses a continuous active learning approach (Di Nunzio et al., 2017) together with a variable threshold based on the geometry of the two-dimensional space of documents (Di Nunzio, 2014). Moreover, the system performs an automatic estimation of the number of documents that need to be read in order to declare the review complete.

In particular, 1) we analyze the results of those topics for which the retrieval system does not achieve a perfect recall; 2) based on this analysis, we perform new experiments to compare the results achieved with the use of either a stemmer or a lemmatizer. This paper is organized as follows: in Section 1.1, we give a brief summary of the use of stemmers and lemmatizers in Information Retrieval; in Section 3, we describe the failure analysis carried out on the CLEF 2017 TAR dataset and the results of the new experiments comparing the use of stemmers vs lemmatizers. In Section 4, we give our conclusions.

1.1 Stemming and Lemmatization

Stemming and lemmatization play an important role in order to increase the recall capabilities of an information retrieval system (Kanis and Skorkovská, 2010; Kettunen et al., 2005). The basic principle of both techniques is to group similar words which have either the same root or the same canonical citation form (Balakrishnan and Lloyd-Yemoh, 2014). Stemming algorithms remove suffixes as well as inflections, so that word variants can be conflated into their respective stems. If we consider the words *amusing* and *amusement*, the stem will be *amus*. On the other hand, lemmatization uses vocabularies and morphological anal-

yses to remove the inflectional endings of a word and to convert it in its dictionary form. Considering the example below, the lemma for *amusing* and *amused* will be *amuse*. Stemmers and lemmatizers differ in the way they are built and trained. Statistical stemmers are important components for text search over languages and can be trained even with few linguistic resources (Silvello et al., 2018). Lemmatizers can be generic, like the one in the Stanford coreNLP package (Manning et al., 2014), or optimized for a specific domain, like BioLemmatizer which incorporates several published lexical resources in the biomedical domain (Liu et al., 2012).

2 System

The system we used in this paper is based on a Technologically Assisted Review (TAR) system which uses a two-dimensional representation of probabilities of a document d being relevant $\mathcal{R}$, or non-relevant, $\mathcal{NR}$ respectively $P(d|\mathcal{R})$ and $P(d|\mathcal{NR})$ (Di Nunzio, 2018).

This system uses an alternative interpretation of the BM25 weighting schema (Robertson and Zaragoza, 2009) by splitting the weight of a document in two parts (Di Nunzio, 2014):

$$P(d|\mathcal{R}) = \sum_{w_i \in d} w_i^{BM25,\mathcal{R}}(tf) \quad (1)$$

$$P(d|\mathcal{NR}) = \sum_{w_i \in d} w_i^{BM25,\mathcal{NR}}(tf) \quad (2)$$

The system uses a bag-of-words approach on the words w_i (either stemmed or lemmatized) that appear in the document and an explicit relevance feedback approach to continuously update the probability of the terms in order to select the next document to show to the user.

In addition, for each topic the system uses a query expansion approach with two variants per topic in order to find alternative and valid terms for the retrieval of relevant documents. Our approach for the query reformulation is based on a linguistic analysis performed by means of the model of terminological record designed in (Vezzani et al., 2018) for the study of medical language and this method allows the formulation of two different query variants. The first is a list of key-words resulting from a systematic semic analysis (Rastier, 1987) consisting in the decomposition of the meaning of technical terms (that is the lexematic or morphological unit) into minimum unit of meaning that cannot be further segmented. The second is a human-readable reformulation using validly attested synonyms and orthographic alternatives as variants of the medical terms provided in the original query. The following examples show our query reformulations given the initial query provided with the CLEF 2017 TAR dataset:

Table 1: CLEF 2017 TAR topics selected for the linguistic failure analysis.

topic ID	# docs shown	# relevant	# missed
CD009579	4000	138	1
CD010339	3000	114	6
CD010653	3320	45	2
CD010783	3004	30	2
CD011145	4360	202	8

- Initial query: *Physical examination for lumbar radiculopathy due to disc herniation in patients with low-back pain*;
- First variant: *Sensitivity, specificity, test, tests, diagnosis, examination, physical, straight leg raising, slump, radicular, radiculopathy, pain, inflammation, compression, compress, spinal nerve, spine, cervical, root, roots, sciatica, vertebrae, lumbago, LBP, lumbar, low, back, sacral, disc, discs, disk, disks, herniation, hernia, herniated, intervertebral*;
- Second variant: *Sensitivity and specificity of physical tests for the diagnosis of nerve irritation caused by damage to the discs between the vertebrae in patients presenting LBP (lumbago)*.

Given a set of documents, the stopping strategy of the system is based on an initial subset (percent p) of documents that will be read and a maximum number of documents (threshold t) that an expert is willing to judge.

3 Experiments

The dataset provided by the TAR in Empirical Medicine Task at CLEF 2017[1] is based on 50 systematic reviews (or topics) conducted by Cochrane experts on Diagnostic Test Accuracy (DTA). For each topic, the set of PubMed Document Identifiers (PIDs) returned by running the

[1] https://goo.gl/jyNALo

query proposed by the physicians in MEDLINE as well as the relevance judgements are made available (Kanoulas et al., 2017). The aim of the task is to retrieve all the documents that have been judged as relevant by the physicians. The results achieved by the participating teams to this task showed that it is possible to get very close to a perfect recall; however, there are some topics for which most of the systems did not retrieve all the possible relevant documents, unless an unfeasible amount of documents is read by the user.

In this paper, i) we present a linguistic and terminological failure analysis of such topics and, based on this analysis, ii) the results of a new set of experiments that compare the use of either a stemmer or a lemmatizer in order to evaluate a possible improvement in the performance in terms of recall. As a baseline for our analyses, we used the source code provided by (Di Nunzio, 2018). The two parameters of the system — the percentage p of initial training documents that the physician has to read, and the maximum number of documents t a physician is willing to read — were set to $p = 500$ and $t = 100, 500, 1000$.

3.1 Linguistic Failure Analysis

In order to select the most difficult topics for the failure analysis, we run the retrieval system with parameters $p = 50\%$ and threshold $t = 1000$ and selected those topics for which the system could not retrieve all the relevant documents, five in total, shown in Table 1. In order to find out why the system did not retrieve all the relevant documents for these topics, we focused on linguistic and terminological aspects both of technical terms in the original query and of the abstracts of missing relevant documents.

We started by reading the abstract of all 19 missing relevant documents and manually selecting technical terms, defined as as all the terms that are strictly related to the conceptual and practical factors of a given discipline or activity (Vezzani et al., 2018), in this case the medical discipline. Then, we compared these terms with those previously identified in the two query variants encoded in the retrieval system. From this comparison, we noticed that most of the relevant terms extracted from the abstracts were not present in the previous two reformulation (a minimum of 0 and a maximum of 8 terms in common), so that some relevant documents in which such terms were present have not been retrieved. By focusing on the morphological point of view, we have been able to categorize such techincal terms in: 1) acronyms; 2) pairs of terms, in particular noun-adjective; 3) triad of terms, in particular noun-adjective-noun.

The category of acronyms is not an unexpected outcome. Medical language is caracterized by an high level of abbreviations and acronyms (Rouleau, 2003) and, in order to retrieve those missing relevant documents, we should have considered all the orthographic variants of a technical term as well as its acronym or expansion according to the case.

Regarding the second and the third category, that is the pairs noun-adjective (e.g.: bile/biliary, pancreas/pancreatic, schizophrenia/schizophrenetic) and the triad of terms noun-adjective-noun (e.g.: psychiatry/psychiatric/psychiatrist), we noticed some problems related to the stemming process. The analysis carried out allowed us to identify numerous cases of understemming, as for example the case of *psychiatry* stemmed as *psychiatri*, *psychiatric* stemmed as *psychiatr* and *psychiatrist* stemmed as *psychiatrist*, all of them belonging to the same conceptual group. The fact that the stemmer recognizes these three words as different suggests us that the conflation of the inflected forms of a lemma in the query expansion procedure may help to retrieve the missed relevant documents.

3.2 Stemming vs Lemmatization

For the reasons explained in the previous section, we decided to perform a new set of experiments on these "difficult" topics to study whether a lemmatization approach can improve the recall compared to the stemming approach. We used the standard algorithms implemented in the two R packages SnowballC[2] and Textstem.[3] Both implements the Porter stemmer (Porter, 1997), while the second uses the TreeTagger algorithm (Schmid, 1999) to select the lemma of a word. To make a fair comparison for the stemming vs lemmatization part of the analysis, in our experiments we did not use any of the two query variants. By reproducing the results presented in (Di Nunzio, 2018), we discovered an issue in the original source code concerning the stemming phase. The R package *tm* for text mining[4] calls the stemming function of the Snow-

[2] https://goo.gl/n3WexD
[3] https://goo.gl/hCLGP8
[4] https://goo.gl/wp859o

ballC with the "english" language instead of the default "porter" stemmer. This caused a substantial difference in the terms produced for the index and those stemmed during the query analysis. For this reason, all our results are significantly higher compared to those presented by (Di Nunzio, 2018) which makes this approach more effective than the original work.

We studied the performance in terms of recall, and precision at 100, 500, and 1000 documents read (p@100, P@500, and P@1000 respectively) for different values of the threshold t. In Table 2, we report in the first column of each value of t the performance of the original experiment compared to our results (only recall is available from (Di Nunzio, 2018)). If we observe the performances on the whole set of test queries, there is no substantial difference between stemming and lemmatization. There is some improvement in terms of recall when threshold $t = 100$, however 85% of recall is usually considered a 'low' score in total recall tasks. Table 3 compares the number of relevant documents missed by the stemming and lemmatization approaches on the difficult topics. The differences between the original experiments and these new experiments are minimal apart from topic CD010339 for which the absence of the two query reformulations led to a worse performance.

4 Final Remarks and Future Work

In this work, we have presented a linguistic failure analysis in the context of medical systematic reviews. The analysis showed that, for those topics where the system does not retrieve all the relevant information, the main issues are related to abbreviations and pairs noun-adjective and the triad of terms noun-adjective-noun. We performed a new set of experiments to see whether lemmatization could improve over stemming but the results were not conclusive. The issues remain the same since the type of relation noun-adjective or noun-adjective-noun, cannot be resolved by a lemmatizer. For this reason, we are currently studying an approach that conflates morphosyntactic variants of medical terms into the same lemma (or 'conceptual sphere') by means of medical terminological records (Vezzani et al., 2018) and the use of the Medical Subject Headings (MesH) dictionary. [5] In this way, we expect that the system will automatically identify all the related forms (such as all the derivative nouns, adjectives or adverbs) of a lemma in order to include them in the retrieval process of potentially relevant documents.

[5] https://meshb.nlm.nih.gov/search

Acknowledgments

The authors would like to thank Sara Bosi and Fiorenza Germana Grilli, students of the Master Degree in Modern Languages for International Communication and Cooperation of the Department of Linguistics and Literary Study of the University of Padua, who helped us in the linguistic failure analysis phase.

References

Vimala Balakrishnan and Ethel Lloyd-Yemoh. 2014. Stemming and lemmatization: A comparison of retrieval performances. *Lecture Notes on Software Engineering*, 2(3):262 – 267.

Giorgio Maria Di Nunzio, Federica Beghini, Federica Vezzani, and Geneviève Henrot. 2017. An Interactive Two-Dimensional Approach to Query Aspects Rewriting in Systematic Reviews. IMS Unipd At CLEF eHealth Task 2. In *Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum, Dublin, Ireland, September 11-14, 2017.*

Giorgio Maria Di Nunzio. 2014. A New Decision to Take for Cost-Sensitive Naïve Bayes Classifiers. *Information Processing & Management*, 50(5):653 – 674.

Giorgio Maria Di Nunzio. 2018. A study of an automatic stopping strategy for technologically assisted medical reviews. In *Advances in Information Retrieval - 40th European Conference on IR Research, ECIR 2018, Grenoble, France, March 26-29, 2018, Proceedings*, pages 672–677.

Lorraine Goeuriot, Liadh Kelly, Hanna Suominen, Aurélie Névéol, Aude Robert, Evangelos Kanoulas, Rene Spijker, João Palotti, and Guido Zuccon, 2017. *CLEF 2017 eHealth Evaluation Lab Overview*, pages 291–303. Springer International Publishing, Cham.

Jakub Kanis and Lucie Skorkovská. 2010. Comparison of different lemmatization approaches through the means of information retrieval performance. In Petr Sojka, Aleš Horák, Ivan Kopeček, and Karel Pala, editors, *Text, Speech and Dialogue*, pages 93–100, Berlin, Heidelberg. Springer Berlin Heidelberg.

Evangelos Kanoulas, Dan Li, Leif Azzopardi, and Rene Spijker, editors. 2017. *CLEF 2017 Technologically Assisted Reviews in Empirical Medicine Overview. In Working Notes of CLEF 2017 - Conference and Labs of the Evaluation forum, Dublin, Ireland, September 11-14, 2017.*, CEUR Workshop Proceedings. CEUR-WS.org.

Table 2: Performance of stemming vs lemmatization for different values of t

	t = 100			t = 500			t = 1000		
	(Di Nunzio, 2018)	stem	lemma	(Di Nunzio, 2018)	stem	lemma	(Di Nunzio, 2018)	stem	lemma
recall	.645	.854	.875	.940	.976	.969	.988	.992	.992
P@100	-	.194	.208	-	.194	.194	-	.194	.208
P@500	-	.113	.108	-	.098	.976	-	.098	.099
P@1000	-	.100	.096	-	.070	.070	-	.071	.071

Table 3: Number of relevant documents missed by the original experiment (see Table 1), the stemming approach (original experiment corrected), and the lemmatization approach.

topic ID	# original	# stem	# lemma
CD009579	1	1	1
CD010339	6	15	16
CD010653	2	1	1
CD010783	2	1	1
CD011145	8	7	9

Kimmo Kettunen, Tuomas Kunttu, and Kalervo Järvelin. 2005. To stem or lemmatize a highly inflectional language in a probabilistic ir environment? *Journal of Documentation*, 61(4):476–496.

Haibin Liu, Tom Christiansen, William A. Baumgartner, and Karin Verspoor. 2012. Biolemmatizer: a lemmatization tool for morphological processing of biomedical text. *Journal of Biomedical Semantics*, 3(1):3, Apr.

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.

Martin F. Porter. 1997. An algorithm for suffix stripping. In Karen Sparck Jones and Peter Willett, editors, *Readings in Information Retrieval*, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

François Rastier. 1987. *Sémantique interprétative*. Formes sémiotiques. Presses universitaires de France.

Stephen E. Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Maurice Rouleau. 2003. La terminologie médicale et ses problèmes. Tribuna, Vol. IV, n. 12.

Helmut Schmid. 1999. Improvements in part-of-speech tagging with an application to german. In *Natural Language Processing Using Very Large Corpora*, pages 13–25. Springer Netherlands, Dordrecht.

Gianmaria Silvello, Riccardo Bucco, Giulio Busato, Giacomo Fornari, Andrea Langeli, Alberto Purpura, Giacomo Rocco, Alessandro Tezza, and Maristella Agosti. 2018. Statistical stemmers: A reproducibility study. In *Advances in Information Retrieval - 40th European Conference on IR Research, ECIR 2018, Grenoble, France, March 26-29, 2018, Proceedings*, pages 385–397.

Federica Vezzani, Giorgio Maria Di Nunzio, and Geneviève Henrot. 2018. TriMED: A Multilingual Terminological Database. In Nicoletta Calzolari (Conference chair), Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 7-12, 2018. European Language Resources Association (ELRA).

Lexical Opposition in Discourse Contrast

Anna Feltracco
Fondazione Bruno Kessler
University of Pavia, Italy
University of Bergamo, Italy
anna.feltracco@gmail.com

Bernardo Magnini
Fondazione Bruno Kessler
Trento, Italy
magnini@fbk.eu

Elisabetta Jezek
University of Pavia
Pavia, Italy
jezek@unipv.it

Abstract

English. We investigate the connection between lexical opposition and discourse relations, with a focus on the relation of contrast, in order to evaluate whether opposition participates in discourse relations.[1] Through a corpus-based analysis of Italian documents, we show that the relation between opposition and contrast is not crucial, although not insignificant in the case of implicit relation. The correlation is even weaker when other discourse relations are taken into account.

Italiano. *Studiamo la connessione tra l'opposizione lessicale e le relazioni del discorso, con attenzione alla relazione di contrasto, per verificare se l'opposizione partecipa alle relazioni del discorso. Attraverso un'analisi basata su un corpus di documenti in italiano, mostriamo che la relazione tra opposizione e contrasto non è cruciale, anche se non priva di importanza soprattutto per i casi di contrasto implicito. La correlazione sembra più debole se consideriamo le altre relazioni del discorso.*

1 Introduction

This paper focuses on lexical opposition and discourse contrast. We define opposition as the relation between two lexical units that contrast with each other with respect to one key aspect of their meaning and that are similar for all the other aspects (e.g. to increase / to decrease, up / down). On the other end, we consider discourse contrast as the relation between two parts of a coherent sequence of sentences or propositions (i.e., discourse arguments) that are in conflict. Both opposition and contrast hold between contrasting elements: the first at the lexical level, the other at the discourse level.

In the following example, a contrast relation is identified between the two arguments in square brackets; two opposite terms are found in the arguments of the relation and are underlined.

(1) [The price of this book increased], while [the price of that one decreased.]

Despite the two relations are *per se* independent, the example shows how opposition can participate in contrast; in fact, the opposites *to increase / to decrease* convey the difference based on which the two mentioned entities (i.e., the books) are compared, leading to a contrast.

Indeed, opposition can be found in the context of other discourse relations (e.g. in the temporal relation "Before the decrease of the demand, an increase of the prices was registered"), and discourse contrast can be conveyed through other strategies (e.g. negation and synonyms "Although the price decreased; the demand did not fall" or incompatibility "She has blue eyes, he has green eyes").

However, our analysis focuses on opposition and contrast, and starts with the observation that both linguistic phenomena involve two elements that are similar in many aspects, but that differ in others (Section 2). This similarity have already been considered by works in the computational field, in which opposition is used as a feature for identifying contrast, and viceversa (Section 3). In this paper, we investigate the behaviour of opposition in the context of a contrast relation adopting a corpus-based approach (Section 4). In particular, we study the opposition-contrast intersection by observing how frequently opposites are found in the arguments of a contrast relation in Contrast-Ita Bank (Feltracco et al., 2017), a corpus anno-

[1] Part of this research has already been published in the first author Ph.D. thesis (Feltracco, 2018).

tated with the discourse contrast relation. We analyze the cases in which the two phenomena co-occur, in order to understand the contribution of opposition to discourse contrast (Section 5). The investigation lead us to enrich Contrast-Ita Bank with lexical opposition. Enlarging our focus, we also investigate the behaviour of opposition in the context of other discourse relations in the corpus, by examining which are the relations that involve pairs of opposites in their arguments (Section 6). Finally, we report our concluding observations and our hint for further work (Section 7).

2 Lexical Opposition and Discourse Contrast

Our definition of opposition in mainly based on the study of Cruse (1986): according to the author, opposition indicates a relation between two terms that *differ along only one dimension of meaning: in respect to all other features, they are identical* (Cruse, 1986, p.197). Examples of opposition are: *to pass / to fail* or *up - down*. In fact, both *to pass / to fail* refer to the result of an examination, but they describe two possible opposite results. Similarly, both *up / down* potentially describe positions with respect to a reference point, the first refers to a higher position, the latter to a lower position.

This definition has some overlap with those proposed for discourse contrast in two of the most important frameworks focused on the study of discourse relations: Rhetorical Structure Theory (Mann and Thompson, 1988) and Segmented Discourse Representation Theory (Asher and Lascarides, 2003). In these theories, the relation of contrast captures cases in which the arguments in the relation *have some aspects in common* (Mann and Thompson, 1988; Carlson and Marcu, 2001), or *have a similar structure* (Asher, 1993), but they *differ in some respect* (i.e., *contrasting themes* (Asher, 1993)) and *are compared with respect to these differences* (Mann and Thompson, 1988). These definitions are consistent with the Penn Discourse Treebank (PDTB) (Prasad et al., 2007) for the sense tag CONTRAST, which is assigned when the arguments of a relation "share a predicate or a property and the difference between the two situations described in the arguments is highlighted with respect to the values assigned to this property" (Prasad et al., 2007, p. 32).

Both opposition and discourse contrast thus involve comparing two elements that are similar in many aspects, but that differ in others; this holds at the lexical level for opposition and at the discourse level for contrast.

3 Opposition and Contrast in NLP

In the area of NLP, the co-occurrence of the opposition and contrast has been considered, for instance, by Roth and Schulte Im Walde (2014), who use what they call *discourse markers* that typically signal a discourse relation, e.g.*but*, for distinguishing paradigmatic relations, including opposition.

Other contributions in the same area use lexical opposition as feature for detecting contrast. As an example, Harabagiu et al. (2006) base the identification of contrast on the opposition relation, given that in some examples "[..] the presence of opposing information contributes more to the assessment of a CONTRAST than the presence of a cue phrase", such as *but* or *although* (Harabagiu et al., 2006).

Marcu and Echihabi (2002) create a system to identify relations of contrast under the hypothesis that some lexical item pairs can "provide clues about the discourse relations that hold between the text span in which the lexical items occur". In a cross-lingual evaluation for English and Swedish, Murphy et al.(2009) show that opposites (*antonyms* in their terminology) are used for different functions: the most common is the one of "creat[ing] or highlight[ing] a secondary contrast within the sentence/discourse".

On the contrary, Spenader and Stulp (2007) give evidence that opposition is not a strong feature for contrast. In particular, they calculate the co-occurrence of opposite adjectives in the contrast relations marked or non-marked by *but* in a corpus. The authors show that opposition is not common in cases of explicit contrast conveyed by *but*, and it is also not very frequent in cases of non-*but* marked contrast. In a similar way, we intend to evaluate whether opposition is a key feature for contrast, or for other discourse relations.

4 Annotating Opposites in Contrast Relations

We carry on our investigation in Contrast-Ita Bank (CIB) (Feltracco et al., 2017)[2], a corpus of 169 Italian documents manually annotated with 372 contrast relations, following the schema proposed

[2] https://hlt-nlp.fbk.eu/technologies/contrast-ita-bank

in the Penn Discourse Treebank. As in the PDTB, the schema in CIB accounts for the identification of *Arg1* and *Arg2*, the two arguments that are compared in a contrast relations. In CIB, two types of contrast are annotated: i) CONTRAST (138 relations), when one the two arguments is similar to the other in many aspects but different in one aspect for which they are compared, and ii) CONCESSION (272 relations), when one argument is denying an expectation that is triggered from the other.[3] CIB accounts for both explicit relations (341) marked by a lexical element (i.e. *connective*, e.g. *but*, *however*) and implicit relations (31).

To evaluate the role of opposition in the context of a contrast, we manually annotated two opposites *opposite1* and *opposite2*, when the former is part of *Arg1* and the latter is part of *Arg2*. For instance, in Example 1 "The price of this book increased" is *Arg1* and "the price of that one decreased" is *Arg2*, and we marked 'increased" as *opposite1* and "decreased" as *opposite2*.

In this manual exercises, we did not limit our annotation to prototypical opposites (Cruse, 1986, p. 262) or to pairs of mono-token words (typically entries of lexical resources), but we manually marked also larger expressions, including cases similar to Example 2.

(2) [Andrew Smith ha rassegnato le dimissioni ieri], nonostante [i tentativi del premier Tony Blair di convincerlo a rimanere]. [4]

In the example, the light-verb construction *rassegnare le dimissioni* (Eng.'to resign') is considered as the opposite of *rimanere* (Eng.'to remain') and the two are found respectively in the two arguments of the contrast relation, conventionally reported in square brackets.

Furthermore, we include in the annotation also 'opposites in context', that is, pairs of terms that are not intuitively considered opposite but are in an opposition relation in the specific context in which they appear, as it happens in Example 3.

(3) [Sul Nuovo Mercato, Tiscali perde lo 0.05% a 2,23], [E. Biscom sale dell'1,09% a 41,44]. [5]

The two terms *perdere* and *salire* (Eng. 'to lose x', 'to fall by x') are semantically opposite in the specific context of Example 3: they are used in their sense of 'loosing (some value)' and 'increasing (of some value)'.

5 Results of the Annotation

We study the connection between opposition and contrast observing the co-occurrence of the two linguistic phenomena and analyzing whether opposition participates in creating contrast.

5.1 Co-occurrence of the two relations

Out of the 372 contrast relations annotated in CIB, we identified a total of 23 cases in which opposites are present in the arguments of a contrast relation[6].

Table 1 shows that opposition is present both when contrast is conveyed explicitly by mean of a connective (as by *nonostante* in Example 2), and when there is no such element (Example 3); however, there is a higher occurrence when the relation is implicit (16% vs 5.2%). With respect to the types of opposition, it occurs both when CONTRAST or CONCESSION have been marked (Examples 3 and 2 respectively), but it is more frequent with the type CONTRAST (9.2% vs 2.5%).

Senses	Types		tot	% over tot
	Explicit	Implicit		
Contrast	7	4	11	9.2%(/102)
Concession	6	0	6	2.5% (/234)
Both	5	1	6	16.6% (/36)
tot	18	5	**23**	
% over tot	5.2%(/341)	16% (/31)		

Table 1: Opposition in discourse contrast in CIB.

5.2 The role of opposition

We conducted a deeper investigation in order to evaluate whether the opposites in the arguments of a contrast relation actually contribute to it.

In Example 4 opposition triggers the contrast relation.

(4) [uno dei due è ricco di cellule staminali], [l' altro ne è povero].[7]

In this case (and in Examples 2 and 3), the contrast relation holds because two entities (e.g. 'one', 'the other') that share a property (i.e. 'to

[3]The presence of one type of relation does not exclude the other.

[4]Eng.:[Andrew Smith resigned yesterday,] despite [Prime Minister Tony Blair's attempts to persuade him to stay.]

[5]Eng.:[On the New Market, Tiscali looses 0.05% to 2.23], [E. Biscom rises by 1.09% to 41.44].

[6]We manually recognized 20 relations; other 3 were identified *ad posteriori* applying the methodology described in Section 6.

[7]Eng.:[one is rich in stem cells],[the other is poor of them.]

have stem cell') are compared with respect to different values that this property takes (i.e. 'to be rich of them', 'to be poor of them'): the different values can be expressed through opposites (i.e. *ricco*/*povero*).

Other examples includes case in which the contrast relation stem from a comparison between two values of a property assigned to the same entity, as happens for the example in Example 5.

(5) Il commercialista [doveva essere il cavaliere bianco chiamato a salvare la Chini] e, invece, [è stato quello che l' ha affossata].[8]

In the example, the contrast arises from the comparison between the opposite roles of the participant: *to save (something) / to ruin (something)*.

Opposition is central for the discourse contrast in these examples. This is not the case for Example 6, for which the opposition does not act as a source for the discourse contrast relation.

(6) [A dispetto degli sforzi della pubblica amministrazione..], [gli investimenti privati in termini di istruzione sono ancora bassi.][9]

In the example, the opposite adjectives *pubblico / privato* (Eng. 'private / public') are attributes of two entities involved: one can say that the participants do have opposite characteristics. However, the contrast relation does not stem from this opposition; rather, it is based on the comparison between the 'positive efforts' on the one hand and the 'low investments' on the other hand.

Out of 23 cases, in 17 opposites are crucial for the contrast relation while in 6 they do not affect the contrast relation. It seems that when opposites appear in the context of a contrast relation they frequently contribute to the phenomena.

We also performed an inter annotator agreement exercises among two annotators to understand whether to distinguish cases in which opposition contributes in conveying the discourse relation (and cases in which they do not) is an easy operation.[10] We register disagreement in 3 cases out of 20, that corresponds to a Dice's coefficient of 85%. After a reconciliation step, in which annotators compared their annotations, and could revise their decisions, two cases were solved, while a third, reported in Example 7 remained.

(7) [A decorrere da domenica 12 entra in vigore il nuovo orario invernale per il servizio extraurbano e la Trento - Malè.] [Da lunedì 13 entra invece in vigore il nuovo orario invernale 2004 / 2005 per il servizio urbano di Trento e Rovereto.] [11]

In this case, one annotator considered that the contrast among the two situations described in the arguments of the discourse relation originates from the opposites *suburban / urban*. Conversely, the other annotator recognized the different dates of entering into force of the two service (i.e. Sunday 12 vs Monday 13) as the source of the resulting discourse contrast.

6 Opposition and Other Discourse Relations

We performed a further analysis evaluating cases of opposites in other discourse relations. We carried on this investigation inspecting the entire CIB corpus and adopting an external resource in which opposites are registered[12]. We automatically retrieved from the corpus pairs of opposites in a windows of 25 token[13]. We retrieved 152 cases that we manually analyzed considering:

- whether the two opposites appear in their opposite sense (e.g. the verbs *andare* / *tornare* are opposite as far as the first verb is not consider as a modal) - data are reported in the second column of Table 2-, and if so:
- whether they are somehow related in the text or not (e.g. in *è subentrato un fatto nuovo, determinato dal fatto che i vincitori del vecchio regime non..* the two opposites properties are of two unrelated entities while in *proposte ufficiali o ufficiose*, the two opposites are in a coordinating relation) - data are

[8] Eng.: The accountant [was supposed to be the white knight designated to save the Chini] and, on the contrary, [he has been the one that ruined it.]

[9] Eng.: [Despite public administration efforts.], [private investments in terms of education are still low.]

[10] One annotator is an author of this paper, the second annotator, who has some familiarity with linguistic tasks, was provided with simple oral instructions through which we ask her to judge the contribution of the opposites when in the context of a contrast relation. We acknowledge Enrica Troiano for collaborating as second annotator.

[11] Eng.: [Starting from Sunday 12 the new winter timetable for the suburban service and for the Trento - Mal enters into force.][From Monday 13 instead the new winter timetable 2004 / 2005 for the urban service of Trento and Rovereto enters into force.]

[12] Dizionario dei Sinonimi e dei Contrari - Rizzoli Editore, http://dizionari.corriere.it/dizionario_sinonimi_contrari

[13] The number was set observing that opposites were found at a maximum distance of 24 tokens in contrast relations.

reported in the third column of Table 2. If the opposites are related:

- whether they are in the arguments of a discourse relation, as in Example 4 - fourth column of the table.

Total	Opposite sense	Related	In Discourse relation
152	100	72	19

Table 2: Opposition in discourse relations.

Results show that in a large number of pairs the two opposites are not actually used in their opposite sense (52 cases = 152 - 100) or are not related in the text (28 cases = 100 - 72). The opposites are found in the arguments of a discourse relation just in 18 cases (11.8 % of the total), suggesting that lexical opposition is not an indicator for the presence of a discourse relation.

A further analysis brought us to investigate also in which discourse relations opposites are involved, following the PDTB classification.[14] We also investigated if opposition is central for these relations. Data are reported in Table 3.

# opp. per relation	discourse relation	# opp. central per relation
7	Comparison.*Contrast*	1
1	Comparison.*Concession*	1
6	Expansion.*Conjunction*	3
3	Expansion.*Level-of-detail*	1
1	Contingency.*Cause*	1
1	Contingency.*Condition*	1
19		8

Table 3: Number of opposition relations in different discourse relations, and their centrality.

From Table 3, we see that opposition co-occurs with different discourse relations, especially Conjunction, but in a more limited number of cases with respect to contrast.[15]

Moreover, comparing the first and the third column of the table, it can be noticed that, as it happens for discourse contrast (see Section 5.2), opposition is not always contributing to the discourse relation itself, meaning that it does not play central role in conveying the relation. As an example, compare Example 8 in which opposition is judged as central, with Example 9 in which it is not.

(8) Sabato [partenza alle 7.01] ed [arrivo alle 19.36.][16]

(9) [..il gruppo ha proseguito l'opera di riorganizzazione societaria], [mettendo un po' d' ordine nelle partecipazioni non legate al core business delle singole controllate..][17]

In the Conjunction relation of Example 8, the two opposite terms indicate the (opposite) events that are coordinated via the conjunction *e*. In Example 9 (a case of EXPANSION.Level-of-detail relation), the two opposites are somehow related (i.e. the *group* is operating for the *singles* subsidiaries), but they are not central for the relation, which is determined by the two events: *proseguire l'opera* and *mettendo [..] ordine*.

7 Conclusion and Further Work

Through the annotation of opposites in the arguments of contrast relations in Contrast-Ita Bank, we aim at providing new insights over the role of opposition in discourse contrast. Overall, we register 23 cases of opposition over 372 contrast relations in our dataset. This number is not high and one we can expect the number to be higher in a larger dataset. However, this limited number suggests that the presence of opposites is not an impacting feature for the identification of contrast relation in the Italian language. It is, however, quite frequent for *implicit* relations, suggesting that the use of opposition can be a strategy to convey contrast when there is a lack of a connective (such as *but* or *however*) that lexicalizes the relation. Moreover, we show that also the co-occurrence of opposition and other discourse relations is low. Despite, in related work opposition has been used as a feature for identifying contrast, the result of our investigation suggests that opposition does not appear to be a strong informative feature and this can possibly lead to a decrease in precision in the process of identifying contrast (i.e., many false positives are expected).

Further and symmetrical work includes the classification of the phenomena that can lead to contrast.

[14]The complete list of the PDTB 3.0 relations can be found in (Webber et al., 2016).

[15]The data for CONTRAST and CONCESSION are part of the ones reported in Table 1, which consider also multi-token expressions and 'opposites in context'.

[16]Eng.: On Saturday, [departure at 7.01] and [arrival at 19.36.]

[17]Eng.: [.. the group has continued the work of corporate reorganization], [putting some order in the shareholdings that not tied to the core business of the single subsidiaries..]

References

Nicholas Asher and Alex Lascarides. 2003. *Logics of conversation*. Cambridge University Press.

Nicholas Asher. 1993. *Reference to Abstract Objects in Discourse*. Kluwer Academic Publishers, Dordrecht.

Lynn Carlson and Daniel Marcu. 2001. Discourse tagging reference manual. *ISI Technical Report ISI-TR-545*, 54:56.

D Alan Cruse. 1986. *Lexical semantics*. Cambridge University Press.

Anna Feltracco, Bernardo Magnini, and Elisabetta Ježek. 2017. Contrast-Ita Bank: A corpus for Italian Annotated with Discourse Contrast Relations. In *Proceedings of the Fourth Italian Conference on Computational Linguistic (CLiC-it 2017)*.

Anna Feltracco. 2018. *Lexical Opposition and Discourse Contrast: A Data-driven Investigation*. Ph.D. thesis, University of Bergamo.

Sanda Harabagiu, Andrew Hickl, and Finley Lacatusu. 2006. Negation, contrast and contradiction in text processing. In *Association for the Advancement of Artificial Intelligence (AAAI)*, volume 6, pages 755–762.

William C Mann and Sandra A Thompson. 1988. Rhetorical Structure Theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Daniel Marcu and Abdessamad Echihabi. 2002. An unsupervised approach to recognizing discourse relations. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 368–375. Association for Computational Linguistics.

M Lynne Murphy, Carita Paradis, Caroline Willners, and Steven Jones. 2009. Discourse functions of antonymy: a cross-linguistic investigation of Swedish and English. *Journal of pragmatics*, 41(11):2159–2184.

Rashmi Prasad, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, Aravind Joshi, Livio Robaldo, and Bonnie L Webber. 2007. The Penn Discourse Treebank 2.0 Annotation Manual.

Michael Roth and Sabine Schulte Im Walde. 2014. Combining word patterns and discourse markers for paradigmatic relation classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 524–530.

Jennifer Spenader and Gert Stulp. 2007. Antonymy in contrast relations. In *Seventh International Workshop on Computational Semantics*, volume 3, page 100.

Bonnie Webber, Rashmi Prasad, Alan Lee, and Aravind Joshi. 2016. A discourse-annotated corpus of conjoined vps. *LAW X*, page 22.

A new Pitch Tracking Smoother based on Deep Neural Networks

Michele Ferro
FICLIT, University of Bologna, Italy
lele.ferro4@gmail.com

Fabio Tamburini
FICLIT, University of Bologna, Italy
fabio.tamburini@unibo.it

Abstract

English. This paper presents a new pitch tracking smoother based on deep neural networks (DNN). The proposed system has been extensively tested using two reference benchmarks for English and exhibited very good performances in correcting pitch detection algorithms outputs.

Italiano. *Questo contributo presenta un programma di* smoothing *del profilo intonativo basato su reti neurali* deep. *Il sistema è stato verificato utilizzando due corpora di riferimento e le sue prestazioni nella correzione degli errori di alcuni algoritmi per l'identificazione del pitch sono decisamente buone.*

1 Introduction

The pitch, and in particular the fundamental frequency - F0 - which represents its physical counterpart, is one of the most relevant perceptual parameters of the spoken language and one of the fundamental phenomena to be carefully considered when analysing linguistic data at a phonetic and phonological level. As a consequence, the automatic extraction of F0 has been a subject of study for a long time and in literature there are many works that aim to develop algorithms able to reliably extract F0 from the acoustic component of the utterances, algorithms that are commonly identified as Pitch Detection Algorithms (PDAs).

Technically, the extraction of F0 is a problem far from trivial and the great variety of methodologies applied to this problem demonstrates its extreme complexity, especially considering that it is difficult to design a PDA that works optimally for the different recording conditions, considering that parameters such as speech type, noise, overlap, etc. are able to heavily influence the performance of this type of algorithms.

Scholars worked hard searching for increasingly sophisticated techniques for these particular cases, although extremely relevant for the construction of real applications, considering solved, or perhaps simply abandoning, the problem of the F0 extraction for the so-called "clean speech". However, anyone who has used the most common programs available for the automatic extraction of F0 is well aware that errors of halving or doubling of the value of F0, to cite only one type of problem, are far from rare and that the automatic identification of voiced areas within the utterance still poses numerous problems.

Every work that proposes a new method for the automatic extraction of F0 should perform an evaluation of the performances obtained in relation to other PDAs, but, usually, these assessments suffer from the typical shortcomings deriving from evaluation systems: they usually examine a very limited set of algorithms, often not available in their implementation, typically considering corpora not distributed, related to specific languages and/or that contain particular typologies of spoken language (pathological, disturbed by noise, etc.) (Veprek, Scordilis, 2002; Wu *et al.*, 2003; Kotnik *et al.*, 2006; Jang *et al.*, 2007; Luengo *et al.*, 2007; Chu, Alwan, 2009; Bartosek, 2010; Huang, Lee, 2012; Chu, Alwan, 2012). There are few studies, among the most recent, that have performed quite complete evaluations that are based on corpora freely downloadable (deCheveigné, Kawahara, 2002; Camacho, 2007; Wang, Loizou, 2012). These studies use very often a single metric in the assessment that measures a single type of error, not considering or partly considering the whole panorama of indicators developed from the pioneering work of Rabiner and colleagues (1976) and therefore, in our opinion, the results obtained seem to be rather partial.

Tamburini (2013) performed an in depth study of the different performances exhibited by several

widely used PDAs by using standard evaluation metrics and well established corpus benchmarks.

Starting from this study, the main purpose of our research was to improve the performances of the best Pitch Detection Algorithms identified in Tamburini (2013) by introducing a post-processing smoother. In particular, we implemented a pitch smoother adopting Keras[1], a powerful high-level neural networks application program interface (API), written in Python and capable of running on top of TensorFlow, CNTK, or Theano.

2 Pitch error correction and smoothing

Typical PDAs are organised into two different modules: the first stage tries to detect pitch frequencies frame by frame and, in the second stage, the pitch candidates or probabilities are connected into pitch contours using dynamic programming techniques (Bagshaw, 1994; Chu, Alwan, 2012; Gonzalez, Brookes, 2014) or hidden Markov models (HMMs) (Jin, Wang, 2011; Wu *et al.*, 2003).

These techniques are, however, not completely satisfactory and various kind of errors remain in the intonation profile. That is why in the literature we can find various studies aiming at proposing pitch profile smoothers. Some works try to correct intonation profile by applying traditional techniques (Zhao *et al.*, 2007; So *et al.*, 2017; Jlassi *et al.*, 2016), while few others (see for example (Kellman, Morgan, 2016; Han, Wang, 2014)) are based on DNN (either Mulity-Layer Perceptrons or Elman Recurrent Neural Networks).

The pitch smoother we propose is based on recurrent neural networks in order to process the entire sequence of raw pitch values computed by the various PDAs and trying to correct it by removing, mainly, halving/doubling errors and other kind of glitches that could appear in raw pitch profiles.

At the input layer we inject one-hot vectors representing the frame pitch value in the interval 0-499Hz as detected by the PDA. We kept the pitch frame size required by each PDA imposing only a frame shift of 0.01 sec for every PDA. With regard to the hidden layer we employed a bidirectional Long-Short-Term Memory (LSTM) with 100 neurons for each direction. They are joined together and inserted into a TimeDistributed wrapper layer so that one value per timestep could be predicted (instead getting one value for each sequence) given the full sequence of one-hot vectors provided as input.

At the output softmax layer we expect to get a probability distribution for the pitch values in the same interval 0-499Hz, considering the most likely one as the actual network prediction. This means that the network input and output layers contain 500 neurons each.

3 Experiments setup

3.1 Tested PDAs

We chose the three PDAs exhibiting the best performances in Tamburini (2013), namely RAPT, SWIPE' and YAAPT. Even though they were originally developed as MATLAB functions, we decided to adopt the corresponding Python implementations.

The primary purpose in the development of RAPT (A Robust Algorithm for Pitch Tracking) (Talkin, 1995) was to obtain the most robust and accurate estimates possible, with little thought to computational complexity, memory requirements or inherent processing delay. This PDA is designed to work at any sampling frequency and frame rate over a wide range of possible F0, speaker and noise condition. For the determination of the pitch profile, a Normalized Cross-Correlation Function (NCCF) is used and each candidate of F0 is estimated thanks to dynamic programming techniques. The Python implementation is available at `http://sp-tk.sourceforge.net/`.

SWIPE (The Sawtooth Inspired Pitch Estimator) (Camacho, 2007) improves the performance of pitch tracking adopting these measures: it avoids the use of the logarithm of the spectrum, it applies a monotonically decaying weight to the harmonics, then the spectrum in the neighbourhood of the harmonics and middle points between harmonics are observed and smooth weighting functions are used. We adopted SWIPE', a variant of this PDA that only uses the main harmonics for pitch estimation, implemented in Python and it is available again at `http://sp-tk.sourceforge.net/`.

The YAAPT (Yet Another Algorithm for Pitch Tracking) (Zahorian, Hu, 2007) is a fundamental frequency (Pitch) tracking algorithm, which is designed to be highly accurate and very robust for both high quality and telephone speech. In gen-

[1] https://keras.io/

eral, a preprocessing step is used to create multiple versions of the signal. Consequently, spectral harmonics correlation techniques (SHC) and a Normalized Cross-Correlation Function (NCCF, as in RAPT) are adopted. The final profile of F0 is estimated thanks to dynamic programming techniques. For our experiments we employed pYAAPT, a Python implementation available at `http://bjbschmitt.github.io/AMFM_decompy/pYAAPT.html`.

3.2 Gold Standards

The evaluation tests were based on two English corpora considered as gold standards, both freely available and widely used in literature for the evaluation of PDAs:

- Keele Pitch Database (Plante *et al.*, 1995): it is composed of 10 speakers, 5 males and 5 females, who read, in a controlled environment, a small balanced passage (the 'North Wind story'). The corpus contains also the output of a laryngograph, from which it is possible to accurately estimate the value of F0.

- FDA (Bagshaw *et al.*, 1993): it is a small corpus containing 5' of recording divided into 100 utterances, read by two speakers, a male and a female, particularly rich in fricative sound, nasal, liquid and glide, sounds particularly problematic to be analysed by the PDAs. Also in this case the gold standard for the values of F0 is estimated starting from the output of the laryngograph.

3.3 Evaluation metrics

Proper evaluation mechanisms have to introduce suitable quantitative measures of performance that should be able to grasp the different critical aspects of the problem under examination. In Rabiner *et al.* (1976) a de facto standard for PDA assessment measures is established, a standard used by many others after him (e.g. (Chu, Alwan, 2009)). If $E_{voi \to unv}$ and $E_{unv \to voi}$ respectively represent the number of frames erroneously classified between voiced and unvoiced and vice versa, while E_{f0} represents the number of voiced frames in which the pitch value produced by the PDA differs from the gold standard for more than 16Hz, then we can define:

- Gross Pitch Error:

$$GPE = E_{f0}/N_{voi}$$

- Voiced Detection Error:

$$VDE = (E_{voi \to unv} + E_{unv \to voi})/N_{frame}$$

where N_{voi} is the number of voiced frames in the gold standard and N_{frame} is the number of frames in the utterance. These indicators, taken individually or in pairs, have been used in a large number of works to evaluate the performance of PDAs. The two indicators, however, measure very different errors; it is possible to measure the performance using only one indicator, usually GPE, but it evaluates only part of the problem and hardly provide a faithful picture of PDA behaviour. On the other hand, considering both measures leads to a difficult comparison of the results.

To try to remedy these problems, Lee and Ellis (2012) have suggested slightly different metrics, which allow the definition of a single indicator:

- Voiced Error:

$$VE = (E_{f0} + E_{voi \to unv})/N_{voi}$$

- Unvoiced Error:

$$UE = E_{unv \to voi}/N_{unv}$$

- Pitch Tracking Error:

$$PTE = (VE + UE)/2$$

where N_{unv} is the number of unvoiced frames contained in the gold standard. However, trying to interpret the results obtained by a PDA in light of the PTE measurement is rather complex: it is not immediate to identify from the obtained results the most relevant source of errors.

In the light of what has been said so far, it seems appropriate to introduce a new measure of performance that is able to easily capture the performance of a PDA in a single, clear indicator that considers all types of possible errors to be equally relevant. So, following Tamburini (2013), we adopt, the Pitch Error Rate as performance metric, defined as:

$$PER = (E_{f0} + E_{voi \to unv} + E_{unv \to voi})/N_{frame}$$

This measure sum all the types of possible errors without privileging or reducing the contribution of any component and allowing a simpler interpretation of the obtained outcomes.

4 Results

We repeated the same experiments as in Tamburini (2013) with the Python implementations of the chosen algorithms (See Table 1) in order to derive common baselines. We also computed the median of the values as in Tamburini (2013) as a simple smoothing method. As in the cited work, it emerges quite clearly that the combination of different algorithms with the median method improves the PER results.

Keele Pitch Database				
PDA	PER	E_{f0}	$E_{voi \to unv}$	$E_{unv \to voi}$
pYAAPT	0.14056	0.04278	0.04411	0.05366
RAPT	0.12596	0.03789	0.05252	**0.03554**
SWIPE'	0.14236	0.02762	0.06985	0.04488
Median	**0.08814**	**0.02656**	**0.03359**	0.03564
FDA Corpus				
PDA	PER	E_{f0}	$E_{voi \to unv}$	$E_{unv \to voi}$
pYAAPT	0.11912	0.03023	**0.03399**	0.0549
RAPT	0.09533	0.01978	0.03438	0.04116
SWIPE'	0.10594	**0.01385**	0.04773	0.04434
Median	**0.10182**	0.02537	0.03686	**0.03917**

Table 1: The experiments in Tamburini (2013) reproduced using the considered PDA python implementation.

After the influential paper from Reimers and Gurevych (2017) it is clear to the community that reporting a single score for each DNN training session could be heavily affected by the system initialisation point and we should instead report the mean and standard deviation of various runs with the same setting in order to get a more accurate picture of the real systems performances and make more reliable comparisons between them.

In order to carry out the experiments with our new pitch smoother we had to split our datasets into training/validation/test set. For the final evaluation of our pitch smoother, we considered only the PER measure. This metric was computed for each epoch during the training phase for all subsets in order to determine the stopping epoch when we get the minimum PER on the validation set. We performed 10 runs for each experiment computing means, standard deviations and significance tests.

We also tested our pitch smoother on a mixed configuration joining our datasets and adopting the same procedures.

Table 2 shows all the obtained results. The proposed system always exhibits the best results in any experiment with relevant performance gains with respect to the PDAs base outputs. All the differences resulted highly significant when applying a t-test. Given the very small standard deviation in all the experiments we can conclude that, in this case, the initialisation point did not affect the network performances too much.

Keele Pitch Database			
PDA	PDA PER	Smoother PER μ	Smoother PER σ
pYAAPT	0.14056	**0.05458**	0.00157
RAPT	0.12596	**0.08726**	0.00193
SWIPE'	0.14236	**0.09666**	0.00298
FDA Corpus			
PDA	PDA PER	Smoother PER μ	Smoother PER σ
pYAAPT	0.11912	**0.06530**	0.00277
RAPT	0.09533	**0.06698**	0.00133
SWIPE'	0.10594	**0.07205**	0.00215
Mixed Keele+FDA Corpus			
PDA	PDA PER	Smoother PER μ	Smoother PER σ
pYAAPT	0.06951	**0.05415**	0.00128
RAPT	0.09859	**0.07341**	0.00133
SWIPE'	0.08758	**0.08288**	0.00163

Table 2: PER mean (μ) and standard deviation (σ) obtained by the proposed pitch profile smoother. One sample t-test significance test returns $p \ll 0.001$ for all experiments. N.B.: Even if the number of experiments is small (10), the power analysis of the t-tests is always equal to 1.0 showing maximum t-test reliability.

5 Conclusions

This paper presented a new pitch smoother based on deep neural networks that obtained excellent results when evaluated using standard benchmarks for English and evaluation metrics proposed in the literature.

Future works could regard the intermixing of various corpora in different languages in order to test the possibility of deriving a pitch smoother able to properly work without caring about language and, possibly, specific corpora and language registers.

Acknowledgements

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Ti-

tan Xp GPU used for this research.

References

Bartosek, J. 2010 Pitch Detection Algorithm Evaluation Framework *In Proceedings of 20th Czech-German Workshop on Speech Processing*, Prague, 118123.

Bagshaw, P.C. 1994 *Automatic prosodic analysis for computer-aided pronunciation teaching*, PhD Thesis, University of Edimburgh.

Bagshaw, P.C. and Hiller, S.M. and Jack, M.A. 1993 Enhanced pitch tracking and the processing of f0 contours for computer aided intonation teaching, *Proceedings of Eurospeech '93*, Berlin, 1003–1006

Camacho A. 2007 SWIPE: A sawtooth waveform inspired pitch estimator for speech and music. *PhD Thesis, University of Florida*.

Chu, W. and Alwan A. 2009 Reducing F0 frame error of F0 tracking algorithms under noisy conditions with an unvoiced/voiced classification frontend *In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP2009*, 39693972.

Chu, W. and Alwan, A. 2012. SAFE: A statistical approach to F0 estimation under clean and noisy conditions. *IEEE Trans. Audio, Speech, Lang. Process.*, 20(3):933–944.

de Cheveigné A. and Kawahara H. 2002 YIN, a fundamental frequency estimator for speech and music *Journal of the Acoustical Society of America*, 111, 191730.

Gonzalez, S. and Brookes, M. 2014. PEFAC-A pitch estimation algorithm robust to high levels of noise. *IEEE Trans. Audio, Speech, Lang. Process.*, 22(2):518–530.

Han, Kun and Wang, DeLiang 2014. Neural Network Based Pitch Tracking in Very Noisy Speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 22(12):2158–2168.

Huang, F. and Lee, T. 2012 Robust Pitch Estimation Using l1-regularized Maximum Likelihood Estimation. *In Proceedings of 13th Annual Conference of the International Speech Communication Association Interspeech 2012*, Portland (OR).

Jang, S.J. and Choi, S.H. and Kim, H.M. and Choi, H.S. and Yoon Y.R. 2007 Evaluation of performance of several established pitch detection algorithms in pathological voices. *In Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society - EMBC*, Lyon, 620623.

Jin, Z. and Wang, L. 2011. HMM-based multipitch tracking for noisy and reverberant speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 19(5):1091–1102.

Jlassi, Wided and Bouzid, Aicha and Ellouze, Noureddine 2016 A new method for pitch smoothing, *2nd International Conference on Advanced Technologies for Signal and Image Processing*, Monastir, Tunisia, 657–661.

Kellman. M. and Morgan, N. 2017 Robust Multi-Pitch Tracking: a trained classifier based approach, *ICSI Tchnical Report*, Berkeley, CA.

Kotnik, B. and Höge, H. and Kacic, Z. 2006 Evaluation of Pitch Detection Algorithms in Adverse Conditions *In Proceedings of Speech Prosody 2006*, Dresden, PS2883.

Lee, B.S. and Ellis, D. 2012 Noise Robust Pitch Tracking by Subband Autocorrelation Classification *In Proceedings of 13th Annual Conference of the International Speech Communication Association Interspeech 2012*, Portland (OR).

Luengo, I., Saratxaga, I., Navas, E., 2007 Evaluation of Pitch Detection Algorithm under Real Conditions. *In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, Honolulu, Hawaii, 4, 10571060.

Plante, F. and Ainsworth, W.A. and Meyer, G. 1995 A Pitch Extraction Reference Database. *In Proceedings of Eurospeech95*, Madrid, 837840.

Rabiner, L.R. and Cheng, M.J. and Rosenberg, A.E. and McGonegal C.A. 1976 A Comparative Performance Study of Several Pitch Detection Algorithms. *IEEE Transaction on Acoustics, Speech and Signal Processing*, 24, 399418.

Reimers, Nils and Gurevych, Iryna. 2017 Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, 338–348.

So, YongJin and Jia, Jia and Cai, LianHong. 2012 Analysis and Improvement of Auto-correlation Pitch Extraction Algorithm Based on Candidate Set, In Zhihong Q., Lei C., Weilian S., Tingkai W., Huamin Y. (eds) *Recent Advances in Computer Science and Information Engineering: Volume 5*, Springer Berlin Heidelberg, 697–702.

Talkin D. 1995 A robust algorithm for pitch tracking (RAPT). In Kleijn W.B., Paliwal, K.K. (eds) *Speech Coding and Synthesis*, New York: Elsevier, 495518.

Tamburini, Fabio 2013 Una valutazione oggettiva dei metodi pi diffusi per l'estrazione automatica della frequenza fondamentale. *In Atti dell IX Convegno Nazionale dell'Associazione Italiana*

di Scienze della Voce (AISV2013), Bulzoni:Roma, 427–434.

Veprek, P. and Scordilis, M.S. 2002 Analysis, enhancement and evaluation of five pitch determination techniques. *Speech Communication*, 37, 249270.

Wang, D. and Loizou, P.C. 2012 Pitch Estimation Based on Long Frame Harmonic Model and Short Frame Average Correlation Coefficient. *In Proceedings of 13th Annual Conference of the International Speech Communication Association Interspeech 2012*, Portland (OR).

Wu, M. and Wang, L. and Brown G.J. 2003. A multipitch tracking algorithm for noisy speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 11(3):229–241.

Zahorian, S.A. and Hu, H. 2008 A Spectral/temporal method for Robust Fundamental Frequency Tracking. *Journal of the Acoustical Society of America*, 123, 45594571.

Zhao, Xufang and O'Shaughnessy, Douglas and Minh-Quang, Nguyen. 2007 A Processing Method for Pitch Smoothing Based on Autocorrelation and Cepstral F0 Detection Approaches, *Proceedings of the International Symposium on Signals, Systems and Electronics*, Montreal, Canada, 59–62

Using and evaluating TRACER for an *Index fontium computatus* of the *Summa contra Gentiles* of Thomas Aquinas

Greta Franzini, Marco Passarotti
Università Cattolica del Sacro Cuore
{greta.franzini,marco.passarotti}@unicatt.it

Maria Moritz, Marco Büchler
Georg-August-Universität Göttingen
{mmoritz,mbuechler}@etrap.eu

Abstract

English. This article describes a computational text reuse study on Latin texts designed to evaluate the performance of TRACER, a language-agnostic text reuse detection engine. As a case study, we use the *Index Thomisticus* as a gold standard to measure the performance of the tool in identifying text reuse between Thomas Aquinas' *Summa contra Gentiles* and his sources.

Italiano. *Questo articolo descrive un'analisi computazionale effettuata su testi latini volta a valutare le prestazioni di TRACER, uno strumento "language-agnostic" per l'identificazione automatica del riuso testuale. Il caso studio scelto a tale scopo si avvale dell'Index Thomisticus quale gold standard per verificare l'efficacia di TRACER nel recupero di citazioni delle fonti della Summa contra Gentiles di Tommaso d'Aquino.*

1 Introduction

Thomas Aquinas (1225-1274) was a prolific medieval author from Italy: his 118 works, known as the *Corpus Thomisticum*, amount to 8,767,883 words (Portalupi, 1994, p. 583) and discuss a variety of topics, ranging from metaphysical to legal, political and moral theory (Kretzmann and Stump, 1993). The web of references to biblical, ecclesiastical and classical literature that stretches the whole *Corpus Thomisticum* speaks to daunting erudition. In the late 1940s, Humanities Computing pioneer Father Roberto Busa (1913-2011) spearheaded a scholarly effort, known as *the Index Thomisticus*, to manually annotate reuse, both *explicit* (i.e., explicitly introduced by Aquinas as a quote) and *implicit* (i.e., reference to works without quotation), in the texts of Thomas Aquinas (Busa, 1980). Four decades later, Portalupi noted:

> Ancora più difficile sarà [...] il tentativo di confrontare automaticamente tutto Tommaso con tutti i testi di uno o più autori, per rintracciare in modo globale la presenza implicita di una fonte. Per fare questo occorrerebbe che si verificassero due condizioni: in primo luogo, gli autori di cui si studiano le presenze implicite in Tommaso dovrebbero essere informatizzati e interrogabili nella totalità delle loro opere; in secondo luogo, bisognerebbe disporre di un software molto potente e raffinato. (Portalupi, 1994, p. 583) [1]

Today, a once visionary task is conceivable, giving way to studies such as the present, which poses the following research question: to which extent can historical text reuse detection (HTRD) software detect explicit and implicit text reuse in the writings of Thomas Aquinas? To this end, we test the performance of TRACER, a text reuse detection framework, for the creation of an *Index fontium computatus* (a computed index of text reuse). The *Summa contra Gentiles* (ScG) was chosen as a case study because the critical edition used for the *Index Thomisticus*, the 1961 Marietti *Editio Leonina* (Gauthier et al., 1882), is still in use today and because an ongoing treebanking effort of the text will, in future, provide us with the linguistic data needed to further refine the experiments described here (Passarotti, 2011).

1. Our English translation reads: 'It will be even harder to automatically compare all of Thomas against all of the texts of one or multiple authors to check for the presence of implicit sources. Such a task would only be possible under two conditions: firstly, the texts of the authors quoted by Thomas would have to be digitised and searchable in their entirety; secondly, one would need very powerful and sophisticated software'.

2 Related Work

2.1 The significance of text reuse

Text reuse (TR) can be summarily described as the written repetition or borrowing of text and can take different forms. Büchler et al. (2014) separate *syntactic* TR, such as (near-)verbatim quotations or idiomatic expressions, from *semantic* TR, which can manifest itself as a paraphrase, an allusion or other loose reproduction. The study of quotation is key to any philological examination of a text, as it is not only indicative of the intellectual and cultural endowment of an author, but may shed light on the sources used, the relation between works and literary influence. Crucially, quotations may also preserve text that is now lost, thus facilitating efforts of textual reconstruction. [2] Owing to the magnitude of the task, the publication of a work's complete index of references, conventionally known as *Apparatus fontium* or *Index scriptorum*, is rare (Portalupi, 1994, p. 582).

2.2 Text reuse in Thomas Aquinas

Like many of his Christian predecessors, Aquinas' body of work teems with references to secular and Christian literature alike. In the ScG (1259-1265) Aquinas cites 170 works both explicitly and implicitly (Gauthier et al., 1882, Vols. IV-XV). Explicit quotations provide information about the source text and the author and/or work, and can either be direct or indirect (Gauthier et al., 1882, vol. XVI, pp. XVI-XXII). Implicit reuses, in the ScG and in general, are more elusive, as they are almost never syntactically nor lexically-faithful to the original text, thus making them hard for both machines and humans to spot (Portalupi, 1994, p. 582). [3] Durantel notes that Aquinas' tendency in TR is to borrow only what is necessary to fit the flow of his narrative without significant semantic or syntactic deviation from the original (Durantel, 1919, p. 63). And yet, Pelster's observation on Aquinas' paraphrastic reuse of Aristotle might suggest greater deviation (Pelster, 1935, p. 331). [4]

Roberto Busa's effort in the late 1940s resulted in the creation of the *Index Thomisticus*, a manually-lemmatised version of Thomas Aquinas' *opera omnia* (Jones, 2016). Among the annotations, the *Index Thomisticus* tags tokens forming explicit quotations as `QL` if literal (*ad litteram*) and `QS` if a paraphrase (*ad sensum*), and tokens forming *implicit* quotations as `QR` to indicate a reference or citation alluding to another text. An example quotation in the ScG containing a mixed annotation is:

> [...] ratio `(QL)` vero `(QL)` significata `(QL)` per `(QL)` nomen `(QL)` est `(QL)` definitio `(QL)` secundum `(QR)` philosophum `(QR)` in `(QR)` IV `(QR)` Metaph. `(QR)` [5]

The `(QL)` portion of this example contains the literal quote, while the second `(QR)` portion provides the reference.

2.3 Historical text reuse detection

HTRD is a Natural Language Processing (NLP) task aimed at identifying syntactic and semantic TR in historical sources. The computational analysis of historical languages is particularly challenging as tools at our disposal are often trained on a synchronic rather than diachronic state of a language [6] and on controlled textual corpora. Eger et al. (2015) and Passarotti (2010) tested the performance of seven different taggers, including TreeTagger (Schmid, 1994), for different training sets and tag-sets of medieval (church) Latin texts showing accuracies tightly below 96% and 96.75% for PoS-tagging, and around 90% and 89.90% for morphological analysis, respectively. These results have yet to be generalised to other variants of Latin and can be improved upon with the provision of additional training corpora, treebanked and semantically-tagged, the creation of corpora containing intertexts, or with the expansion of lexical resources, such as the *Latin WordNet* (Minozzi, 2017, p. 130).

The extent to which the limitations of these resources and taggers (e.g., correct resolution of homographs) affect HTRD tools, including *Tesserae* (Coffee et al., 2013), *Passim* (Smith et al., 2015) [7] and *TRACER* (Büchler, 2013) is not yet

2. One notable example is the fragmentary survival of Alexandrian scholarship at the hands of Roman philologists (who wrote commentaries known as *scholia*) and grammarians (Turner, 2014, p. 16).

3. For problems with implicit quotations, see (Haverfield, 1916, p. 197) and (Fowler, 1997, p. 15). For automatic allusion detection, see (Bamman and Crane, 2008).

4. "Da Thomas die Schriften des Aristoteles [...] gewöhnlich nur dem Gedanken nach, nicht wörtlich anführt." In English: 'Since Thomas usually quotes paraphrastically, not literally.'

5. Book 1, chap. 12, n. 4. Our English translation reads: '[...] according to the philosopher in Metaph. IV, the meaning of a name is its definition'.

6. See Janda and Joseph (2005) for the dichotomy.

7. `https://github.com/dasmiq/passim`

fully understood. Reasons for this are the field's lack of progress caused by "inconsistent standards and the scattering of insights across publications" (Coffee, 2018), the general failure of HTRD studies to publish negative results, and the quasi-absence of gold standards for testing. To our knowledge, the only projects to have published computed results from intertextual studies on historical sources are the *Proteus Project* (English and Latin) (Yalniz et al., 2011), the *Chinese Text Project* (early Chinese) (Sturgeon, 2017), *Commonplace Cultures* (English and Latin) (Gladstone and Cooney, forthcoming), *SHEBANQ* (Hebrew) (Naaijer and Roorda, 2016), *Samtla* (Search and Mining Tools for Language Archives) (language-independent) (Harris et al., 2018), and *Tesserae* (Latin), but of these only the latter discloses tool configurations.

3 Methodology

3.1 Gold Standard

To facilitate the classification of automatically-detected reuse, all `QL`-, `QS`- and `QR`-annotated tokens were extracted from the *Index Thomisticus*. Of the total 24,416 sentences constituting the ScG, the 7,396 (30.29%) containing any combination of `QL`, `QS` and `QR` were stored in a tabular file, which we define as the *Index Thomisticus Gold Standard* of TR (hereafter IT-GS). The number of sentences containing only `QL` tokens (1,139) compared to that of sentences containing only `QS` tokens (2,270) corroborates expert assertions about Aquinas' paraphrastic style of TR.

3.2 Text acquisition and preparation

For the sake of processing efficiency, out of the ScG's 170 source works we began with a set of five readily available texts. These are *Philosophiae Consolationis* and *De Trinitate* of Boethius, *De Deo Socratis* of Apuleius, Cicero's *De Divinatione* and the Moerbeke Latin translation of Aristotle's *Metaphysica*. The texts were acquired from different sources and cleaned of all paratextual information. The clean texts were then segmentised by sentence, PoS-tagged and lemmatised with the TreeTagger Brandolini parameter file (with an average accuracy of 93.72%), whose tag-set provides the degree of granularity needed in this experiment. [8] Finally, a script was used to format sentences to TRACER requirements.

3.3 Text reuse detection with TRACER

The HTRD on this corpus was performed (server-side) with TRACER, a language-agnostic framework comprising hundreds of information retrieval (IR) algorithms designed to work with historical and modern languages alike. [9] TRACER is a Java command-line tool driven by an XML configuration file, which users can modify to fit their detection needs. TRACER follows a six-step architecture, [10] which demystifies the detection process by storing the computed output of each step on the disk so that users can more easily follow and locate errors in the processing chain, if any. TRACER is resilient to OCR-noise and capable of detecting both (near-)verbatim quotations and looser forms of TR. The detection of paraphrase requires the use of linguistic resources to help TRACER match a word against its synsets and an inflected form against its base-form. For synonym detection, we extracted synonymous relations from the Latin WordNet. TR identified with TRACER was manually compared against the IT-GS to separate the True (TP) from the False Positives (FP), and to identify False Negatives (FN).

4 Results

4.1 Philosophiae Consolationis

To detect both verbatim quotations and paraphrase, TRACER was optimised for recall over precision and configured to work with single words as features, to ignore the top 20% most frequent words, [11] to link text pairs with a minimum overlap of 5 features, [12] to expand the query to synonyms, and to return only those aligned text pairs presenting an overall sentence similarity of at least 50%. [13] Of the eight reuses indicated in

8. The Brandolini tag-set was manually mapped against that of Morpheus (Crane, 1991), which TRACER uses as a reference. Ambiguously-lemmatised word forms were not disambiguated.

9. `https://doi.org/21.11101/0000-0007-C9CA-3`

10. The six steps are: *Preprocessing*, *Featuring*, *Selection*, *Linking*, *Scoring* and *Postprocessing*.

11. The parameter, known as *feature density*, is a language-independent measure used to decontaminate the texts and to contain the number of results based on chance repetition; an 80% feature density means that TRACER ignores or removes the most frequent types that cover 20% of the tokens.

12. For a 24k sentence corpus such as this, an overlap of 5 is statistically significant (Büchler, 2013, p. 134).

13. The value was chosen on the basis of previous experiments as a good trade-off between precision and recall. The similarity measure used is Broder's *containment*, which is particularly suited to documents or sentences of uneven

the *Editio Leonina*, we were unable to precisely locate one as it alludes to four paragraphs of text; [14] of the remaining seven, as shown in Figure 1, TRACER identified three (42%). Upon close inspection, two FNs were affected by the 20% threshold of feature removal, for example:

Boethius 1.4.105 *Unde haud iniuria tuorum quidam familiarium* ***quaesivit****: "****Si*** *quidem* ***deus****", inquit, "****est****,* ***unde*** ***mala*** *?* [15]

Aquinas 3.71.10 , *introducit quendam philosophum* ***quaerentem****:* ***si deus est****,* ***unde*** ***malum*** *?* [16]

Here, the tokens *si*, *est* and *unde* were ignored as they fell within the pool of the 20% most frequent words removed.

One reuse was successfully identified on the basis of feature overlap but did not amount to a 50% sentence similarity; and the fourth reuse could not be identified because of a missing synonymous relation in the Latin WordNet (i.e., *gaudium-beatitudo*) [17] and its insufficient feature overlap. The resulting F1-score is $4,6 \cdot 10^{-3}$.

4.2 De Trinitate

Given the results of the previous analysis, for this second investigation the feature removal and the sentence similarity values were lowered to 10% and 40% respectively, thus optimising for even higher recall (10,349 total sentences aligned). Of the four known reuses, TRACER identified three. The 40% similarity threshold was essential to the identification of one reuse (where the score is 0.4375); the FN, which was indeed found on the basis of an eight-word overlap but did not meet the minimum sentence similarity threshold, revealed another missing synonymous relation in the WordNet (i.e., *disciplinatus-eruditus*) [18] and a failed alignment of the variants *temptare* (Boethius) and *tentare* (Aquinas) owing to inconsistent TreeTagger lemmatisation (*tempto* and *tento*, respectively). The F1-score for this analysis was $5,6 \cdot 10^{-4}$.

4.3 De Deo Socratis

This work of Apuleius is quoted twice in the ScG. Of the two reuses, TRACER was able to detect one in full and only parts of the second. The second reuse spans three sentences and is mostly paraphrastic, with only three words annotated in the *Index Thomisticus* as QL (*sunt animo passiva*). [19] To capture the fullest range of reuse diversity, TRACER's feature removal was set to 10%, the overlap to 3 and the overall similarity to 20%. However, as *sunt* (form of the verb *sum* 'to be') is the most frequent word across the texts, TRACER's inbuilt feature removal prevented the detection of the short QL portion of the reuse; the QR+QS portions, on the other hand, were successfully detected. We counted both results as TPs, resulting in an F1-score of $2,6 \cdot 10^{-5}$.

4.4 De Divinatione

The only recorded reuse that Aquinas makes of Cicero's text is implicit and alludes to a block of text, making it difficult to manually pinpoint with precision. To detect as loose a similarity as possible, the TRACER search was cast with the same configuration used in the previous analysis. No reuse, however, was found.

4.5 Metaphysica

The *Editio Leonina* lists 97 reuses of Aristotle's *Metaphysica*. As previously mentioned, Pelster describes Aquinas' reuse of the Latin translation of the *Metaphysica* as more paraphrastic than literal. Our manual examination of the texts and the results of TRACER confirmed this observation, in that we could not manually locate seven reuses (due to their strong allusiveness) and a fault-tolerant TRACER configuration (removal of the top 10% most frequent words, overlap of 3 features and an overall sentence similarity of 40%) yielded 19 TPs only (6 out of 15 QL [20] and 13 out of 75 QR+QS). The F1-score resulting from this analysis is $3,8 \cdot 10^{-4}$.

length (Broder, 1997).

14. This reuse would have doubtless been overlooked by TRACER too owing to the absence of features to compare.

15. Our English translation reads: 'It is not wrong that a certain acquaintance of yours has questioned: 'If in fact God exists,' he asks, 'where is evil from?"

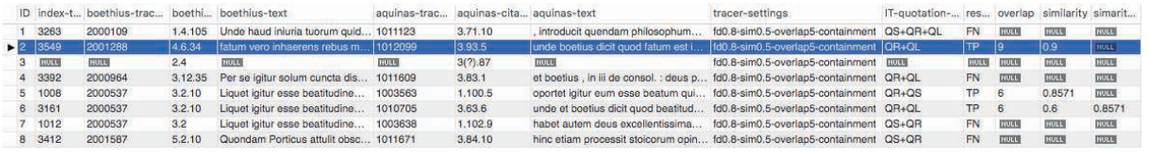
16. Our English translation reads: '(Boethius) introduces a certain philosopher who asks: 'If God exists, where is evil from?'.'

17. Incidentally, this relation is also not mapped in BabelNet (bn:00042905n) nor in ConceptNet (http://conceptnet.io/c/la/gaudium) (as of 8 June 2018).

18. Also not present in neither BabelNet nor ConceptNet.

19. *[daemones] […] sunt animo passiva* or 'demons are emotional in mind' (Jones, 2017, pp. 372-373).

20. The QL quotations in the ScG seem to refer to a different Latin translation than that available to us, which would explain why some instances of QL went undetected.



ID	index-t...	boethius-trac...	boethi...	boethius-text	aquinas-trac...	aquinas-cita...	aquinas-text	tracer-settings	IT-quotation-...	res...	overlap	similarity	simarit...
1	3263	2000109	1.4.105	Unde haud iniuria tuorum quid...	1011123	3.71.10	, introducit quendam philosophum...	fd0.8-sim0.5-overlap5-containment	QS+QR+QL	FN	NULL	NULL	NULL
▶ 2	3549	2001288	4.6.34	fatum vero inhaerens rebus m...	1012099	3.93.5	unde boetius dicit quod fatum est i...	fd0.8-sim0.5-overlap5-containment	QR+QL	TP	9	0.9	NULL
3	NULL	NULL	2.4	NULL	NULL	3(?).87	NULL	fd0.8-sim0.5-overlap5-containment	NULL	NULL	NULL	NULL	NULL
4	3392	2000964	3.12.35	Per se igitur solum cuncta dis...	1011609	3.83.1	et boetius , in iii de consol. : deus p...	fd0.8-sim0.5-overlap5-containment	QR+QL	FN	NULL	NULL	NULL
5	1008	2000537	3.2.10	Liquet igitur esse beatitudine...	1003563	1.100.5	oportet igitur eum esse beatum qui...	fd0.8-sim0.5-overlap5-containment	QR+QS	TP	6	0.8571	NULL
6	3161	2000537	3.2.10	Liquet igitur esse beatitudine...	1010705	3.63.6	unde et boetius dicit quod beatitud...	fd0.8-sim0.5-overlap5-containment	QR+QL	TP	6	0.6	0.8571
7	1012	2000537	3.2	Liquet igitur esse beatitudine...	1003638	1.102.9	habet autem deus excellentissima...	fd0.8-sim0.5-overlap5-containment	QS+QR	FN	NULL	NULL	NULL
8	3412	2001587	5.2.10	Quondam Porticus attulit obsc...	1011671	3.84.10	hinc etiam processit stoicorum opin...	fd0.8-sim0.5-overlap5-containment	QS+QR	FN	NULL	NULL	NULL

FIGURE 1 – For every TRACER analysis, a MySQL table is created to store and manually-evaluate the results against the IT-GS. The evaluation table for *Philosophiae Consolationis* illustrated here contains a wealth of information, including full citation information for both works, the TRACER settings used for the detection task, the *Index Thomisticus* quotation annotations, the result classification (into True Positive and False Negative), as well as the feature overlap and the overall similarity value of the aligned sentences. The reuse in the highlighted row, for instance, was correctly identified by TRACER on the basis of a 9-word overlap and an overall sentence similarity of 90%.

5 Discussion

Our results show that the FNs emerging from the computational analyses were largely caused by Aquinas' paraphrastic and allusive TR style, which at times challenged our own ability to spot similarities, even with the help of the critical edition. The allusions that we could identify generally retain the semantics of the alluded-to texts, thus confirming Durantel's insights. While a number of these negative results were also directly tied to *lacunae* in the Latin WordNet and to inconsistent lemmatisation, the flexibility and methodological transparency of TRACER allowed us to locate error sources and accordingly tune configurations to work around these issues (e.g., by increasing the feature overlap and/or lowering the sentence similarity scoring thresholds). Notwithstanding, TRACER's panlingual feature removal parameter affected the retrieval of shorter instances of reuse, particularly those containing forms of the highly frequent verb *sum*.

The manual evaluation of TRACER results against the IT-GS for the creation of an *Index fontium computatus* was time-consuming, not least because of a number of reference inaccuracies in the critical edition itself (in one case, the reference is off by ten lines). Nevertheless, the creation of the index is proving essential to the assessment of TRACER's fitness for purpose on Latin texts.

As far as the usability of the tool is concerned, TRACER's detection power is offset by its cumbersome setup, which is unfriendly to those who are not familiar with the command line, NLP basics and/or Java (stack traces). This issue is being addressed with the development of a user manual (Franzini et al., 2018).

6 Conclusion

This article describes a computational text reuse study on Latin texts designed to evaluate the performance of TRACER, a language-agnostic IR text reuse detection engine. The results obtained were manually evaluated against a gold standard and are contributing to the creation of an *Index fontium computatus* to both assess TRACER's efficacy and to provide a test-bed against which analogous IR systems can be measured and thus compared to TRACER. Our study shows that despite the known limitations of existing linguistic resources for Latin, the diverse spectrum of paraphrastic reuse encountered and its own language-agnosticism, TRACER is equipped to detect a wide range of explicit text reuse in the ScG, be that short or long, verbatim or paraphrastic, and implicit reuse only if coupled with explicit. To increase the detection accuracy, we are implementing a black/white list to give users the power to control words or multi-word expressions to be ignored or retained in the detection; furthermore, we plan on re-running these analyses with the disambiguated linguistic annotation currently being added to the text of the ScG (Passarotti, 2015) to measure its impact on this particular IR task.

The data used and generated in the current study is available from: `https://github.com/CIRCSE/text-reuse-aquinas`.

Acknowledgments

The authors would like to thank Eleonora Litta for proofreading this article and the anonymous reviewers for their valuable comments. This research was funded by the German Federal Ministry of Education and Research (No. 01UG1409).

References

David Bamman and Gregory Crane. 2008. The Logic and Discovery of Textual Allusion. In *Proceedings of the ACL Workshop LaTeCH - Language Technology for Cultural Heritage Data*. ACL. http://hdl.handle.net/10427/42685.

Andrei Z. Broder. 1997. On the resemblance and containment of documents. In *Proceedings of the Compression and Complexity of Sequences 1997*, SEQUENCES '97, pages 21–29, Washington, DC, USA. IEEE Computer Society. http://dl.acm.org/citation.cfm?id=829502.830043.

Marco Büchler, Philip R. Burns, Martin Müller, Emily Franzini, and Greta Franzini. 2014. Towards a Historical Text Re-use Detection. In Chris Biemann and Alexander Mehler, editors, *Text Mining*, pages 221–238. Springer International Publishing, Cham. http://link.springer.com/10.1007/978-3-319-12655-5_11.

Marco Büchler. 2013. Informationstechnische Aspekte des Historical Text Re-use. PhD Thesis. http://www.qucosa.de/fileadmin/data/qucosa/documents/10851/Dissertation.pdf.

Roberto Busa. 1980. The annals of humanities computing: The Index Thomisticus. *Computers and the Humanities*, 14(2):83–90, October. http://www.jstor.org/stable/30207304.

Neil Coffee, Jean-Pierre Koenig, Shakthi Poornima, Christopher W. Forstall, Roelant Ossewaarde, and Sarah L. Jacobson. 2013. The Tesserae Project: intertextual analysis of Latin poetry. *Literary and Linguistic Computing*, 28:221–228. https://doi.org/10.1093/llc/fqs033.

Neil Coffee. 2018. An Agenda for the Study of Intertextuality. *Transactions of the American Philological Association*, 148:205–223. https://muse.jhu.edu/article/693654.

Gregory Crane. 1991. Generating and Parsing Classical Greek. *Literary and Linguistic Computing*, page 243–245. https://doi.org/10.1093/llc/6.4.243.

Jean Durantel. 1919. *Saint Thomas et le Pseudo-Denis*. Librairie Félix Alcan, Paris. http://archive.org/details/cuasaintthomaset00dura.

Steffen Eger, Tim vor der Brück, and Alexander Mehler. 2015. Lexicon-assisted tagging and lemmatization in Latin: A comparison of six taggers and two lemmatization methods. In *In Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 105–113. http://www.aclweb.org/anthology/W15-3716.

Don Fowler. 1997. On the Shoulders of Giants: Intertextuality and Classical Studies. *Materiali e discussioni per l'analisi dei testi classici*, 39:13–34. http://www.jstor.org/stable/40236104.

Greta Franzini, Emily Franzini, Kirill Bulert, Marco Büchler, and Maria Moritz. 2018. TRACER: A User Manual. https://tracer.gitbook.io/-manual/.

R. A. Gauthier, L. J. Bataillon, A. Oliva, T. de Vio Cajetan, Commissio Leonina, and Dominicans. 1882. *Sancti Thomae Aquinatis Doctoris Angelici Opera Omnia iussu edita Leonis XIII P.M.* Ex Typographia Polyglotta S.C. de Propaganda Fide, Rome.

Clovis Gladstone and Charles Cooney. forthcoming. Opening New Paths for Scholarship: Algorithms to Track Text Reuse in ECCO. *Digitizing Enlightenment*.

Martyn Harris, Mark Levene, Dell Zhang, and Dan Levene. 2018. Finding Parallel Passages in Cultural Heritage Archives. *Journal on Computing and Cultural Heritage*, 11(3):15:1–15:24. http://doi.acm.org/10.1145/3195727.

Francis John Haverfield. 1916. Tacitus during the Late Roman Period and the Middle Ages. *The Journal of Roman Studies*, 6:196–201. https://doi.org/10.2307/296272.

Richard D. Janda and Brian D. Joseph. 2005. On Language, Change, and Language Change – Or, Of History, Linguistics, and Historical Linguistics. In Brian D. Joseph and Richard D. Janda, editors, *The Handbook of Historical Linguistics*, pages 3–181. Wiley-Blackwell, Oxford.

Steven E. Jones. 2016. *Roberto Busa, S. J., and the Emergence of Humanities Computing: The Priest and the Punched Cards*. Routledge, March.

Christopher P. Jones, editor. 2017. *Apuleius. Apologia. Florida. De Deo Socratis*, volume 534 of *Loeb Classical Library*. Harvard University Press, Loeb Classical Library.

Norman Kretzmann and Eleonore Stump, editors. 1993. *The Cambridge Companion to Aquinas*. Cambridge University Press, Cambridge; New York, May.

Stefano Minozzi. 2017. Latin WordNet, una rete di conoscenza semantica per il latino e alcune ipotesi di utilizzo nel campo dell'Information Retrieval. In Paolo Mastandrea, editor, *Strumenti digitali e collaborativi per le Scienze dell'Antichità*, number 14 in Antichistica, pages 123–134. http://doi.org/10.14277/6969-182-9/ANT-14-10.

Martijn Naaijer and Dirk Roorda. 2016. Parallel Texts in the Hebrew Bible, New Methods and Visualizations. *CoRR*, abs/1603.01541. http://arxiv.org/abs/1603.01541.

Marco Passarotti. 2010. Leaving behind the less-resourced status. The case of Latin through the experience of the Index Thomisticus Treebank. In *7th SaLTMiL Workshop on Creation and use of basic lexical resources for less-resourced languages LREC 2010, Valletta, Malta, 23 May 2010*.

Marco Passarotti. 2011. Language Resources. The State of the Art of Latin and the Index Thomisticus Treebank Project. In Marie-Sol Ortola, editor, *Corpus anciens et Bases de données, « ALIENTO*.

Échanges sapientiels en Méditerranée », volume 2, pages 301–320. Presses universitaires de Nancy, Nancy.

Marco Passarotti. 2015. What you can do with linguistically annotated data. From the Index Thomisticus to the Index Thomisticus Treebank. In Vijgen Roszak Piotr, editor, *Reading Sacred Scripture with Thomas Aquinas. Hermeneutical Tools, Theological Questions and New Perspectives*, pages 3–44. Brepols.

F. Pelster. 1935. Die Uebersetzungen der aristotelischen Metaphysik in den Werken des hl. Thomas von Aquin: Ein Beitrag. *Gregorianum*, 16(3):325–348. http://www.jstor.org/stable/23567607.

Enzo Portalupi. 1994. L'uso dell'"Index Thomisticus" nello studio delle fonti di Tommaso d'Aquino: Considerazioni generali e questioni di metodo. *Rivista di Filosofia Neo-Scolastica*, 86(3):573–585. http://www.jstor.org/stable/43062344.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of International Conference on New Methods in Language Processing*, Manchester, UK. http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/tree-tagger1.pdf.

David A. Smith, Ryan Cordell, and Abby Mullen. 2015. Computational Methods for Uncovering Reprinted Texts in Antebellum Newspapers. *American Literary History*, 27(3):E1–E15. http://dx.doi.org/10.1093/alh/ajv029.

Donald Sturgeon. 2017. Unsupervised identification of text reuse in early Chinese literature. *Digital Scholarship in the Humanities*. https://academic.oup.com/dsh/advance-article/doi/10.1093/llc/fqx024/4583485.

James Turner. 2014. *Philology: The Forgotten Origins of the Modern Humanities*. Princeton University Press, Princeton and Oxford.

Ismet Zeki Yalniz, Ethem F. Can, and R. Manmatha. 2011. Partial Duplicate Detection for Large Book Collections. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 469–474. http://doi.acm.org/10.1145/2063576.2063647.

Inter-Annotator Agreement in linguistica: una rassegna critica

Gloria Gagliardi
FICLIT – Università di Bologna, Italy
gloria.gagliardi2@unibo.it

Abstract

Italiano. I coefficienti di Inter-Annotator Agreement sono ampiamente utilizzati in Linguistica Computazionale e NLP per valutare il livello di "affidabilità" delle annotazioni linguistiche. L'articolo propone una breve revisione della letteratura scientifica sull'argomento.

English. *Agreement indexes are widely used in Computational Linguistics and NLP to assess the reliability of annotation tasks. The paper aims at reviewing the literature on the topic, illustrating chance-corrected coefficients and their interpretation.*

1 Introduzione

La costruzione di risorse linguistiche, e più in generale l'annotazione di dati, implicano la formulazione di giudizi soggettivi. La necessità di stabilire fino a che punto tali giudizi siano affidabili e riproducibili ha assunto crescente importanza, fino a rendere le procedure di validazione prassi consolidata. Ciò è avvenuto in linguistica computazionale (LC) con più di 30 anni di ritardo rispetto alla psicometria: già nel 1960 Cohen, in un celebre articolo, scriveva infatti:

> *"Because the categorizing of the units is a consequence of some complex judgment process performed by a 'two-legged meter' [...], it becomes important to determine the extent to which these judgments are reproducible, i.e., reliable."*
>
> (Cohen, 1960: 37)

È convinzione abbastanza diffusa che un alto livello di *Inter-Annotator Agreement* (da ora in poi: I.A.A.) tra gli annotatori sia indice della bontà e della riproducibilità di un paradigma di annotazione. Come sottolinea Di Eugenio:

> *"This raises the question of how to evaluate the 'goodness' of a coding scheme. One way of doing so is to assess its reliability, namely, to assess whether different coders can reach a satisfying level of agreement with each other when they use the coding manual on the same data."*
>
> (Di Eugenio, 2000: 441)

L'assunto di base è dunque che i dati siano considerabili "attendibili" se due o più annotatori sono in accordo nell'individuare un fenomeno linguistico oppure nell'assegnare una categoria all'item in analisi. In tale prospettiva, la *reliability* si configura perciò come prerequisito per dimostrare la validità di uno schema di codifica, e un ampio consenso tra gli annotatori viene assunto a garanzia della precisione intrinseca del processo di annotazione (Warrens, 2010).

> *"The main reason for the analysis of annotation quality is to obtain a measure of the 'trustworthiness' of annotations. […] Only if we can trust that annotations are provided in a consistent and reproducible manner, can we be sure that conclusions drawn from such data are likewise reliable and that the subsequent usage of annotations is not negatively influenced by inconsistencies and errors in the data. Inter-annotator (or inter-coder) agreement has become the quasi-standard procedure for testing the accuracy of manual annotations."*
>
> (Bayerl & Paul, 2011: 700)

In ambito computazionale l'I.A.A. è usato come veicolo per passare dal materiale annotato ad un *gold standard*, ovvero un insieme di dati sufficientemente *noise-free* che serva per *training* e *testing* di sistemi automatici. Di prassi i coefficienti di *agreement* vengono usati per assicurare la bontà della procedura di annotazione e del materiale annotato: un alto livello di I.A.A. fa sì che il fenomeno sia considerato consistente e sistematico, e che la risorsa validata sia idonea per addestrare un sistema automatico che svolga il medesimo compito del linguista.
In realtà, l'idea che l'I.A.A. possa indicare in senso assoluto la qualità del *dataset* come risorsa di riferimento è fallace: due osservatori possono, pur sbagliando entrambi, essere in perfetto accordo nel valutare un evento:

> *"However, it is important to keep in mind that achieving good agreement cannot ensure validity: two observers of the same event may well share the same prejudice while still being objectively wrong."*
>
> (Artstein & Poesio, 2008: 557)

È inoltre opportuno considerare che l'*agreement* raggiunto abitualmente dagli annotatori varia in

relazione al livello di esperienza: l'I.A.A. in gruppi omogenei è comparabile a prescindere dai livelli di esperienza, ma si abbassa qualora vengano formati gruppi misti di esperti e non esperti:

> *"Implicit in discussions of inter-annotator agreement is that coders not only agree on which unit belongs to which category, but that if they agree these decisions are also correct with respect to the phenomenon under scrutiny [...]. In our study, this assumption left us with a dilemma. Our data showed that experts and non-experts could achieve comparable levels of agreement, whereas the average agreement for mixed groups was significantly lower. In other words, experts and novices were equally reliable, yet did not agree with each other."*
>
> (Bayerl & Paul, 2011: 721)

Non tutti i task di annotazione linguistica sono valutabili secondo le stesse procedure; dal punto di vista qualitativo, si possono individuare almeno due tipologie generali (Mathet, Widlöcher, A. & Métivier, 2015):

- "individuazione di unità" o "*unitizing*" (Krippendorff, 1980), in cui l'annotatore, dato un testo scritto o parlato, deve identificare posizione e confine degli elementi linguistici (es. identificazione di unità prosodiche o gestuali, *topic segmentation*);
- "categorizzazione": l'annotatore deve attribuire un *tag* a oggetti linguistici preidentificati (es. *PoS Tagging, Word Sense Disambiguation*).

Il paper si propone di presentare una breve rassegna critica delle metriche utilizzate in questa seconda tipologia di *task*, in particolare ponendo attenzione al calcolo dei coefficienti e alla loro interpretazione.

2 I coefficient di agreement

Adottando la notazione proposta da Artstein & Poesio (2008), ogni studio di I.A.A per i *task* di categorizzazione deve prevedere:

- un insieme di item $\{i \mid i \in I\}$;
- un insieme di categorie assegnabili agli item $\{c \mid c \in C\}$;
- un insieme di annotatori, che assegnano ciascun item ad una categoria $\{r \mid r \in R\}$.

Verrà convenzionalmente indicato con A l'*agreement* e con D il *disagreement*. Allo scopo di illustrare le modalità di calcolo dei coefficienti, è stato creato *ad hoc* un esempio fittizio: la situazione immaginata prevede che due annotatori assegnino 20 item a 3 categorie.

		rater 1			
		*c*1	*c*2	*c*3	tot
rater 2	*c*1	9	2	0	11
	*c*2	0	6	0	6
	*c*3	1	0	2	3
	tot	10	8	2	20

Tab. 1: Esempio di tabella di contingenza

2.1 *Agreement* senza correzione del caso

L'indice più rudimentale è quello percentuale, detto anche **"Index of crude agreement"** (Goodman & Kruskal, 1954) o **"Observed Agreement" (A_o)**: la misura corrisponde, banalmente, al rapporto tra il numero di item su cui i *rater* sono d'accordo ed il numero totale di item. Nell'esempio proposto in tab.1, A_o ha un valore di 0.85.

La misura non solo non tiene in considerazione il ruolo che potrebbe giocare il caso, per cui i *rater* potrebbero trovarsi in accordo "tirando ad indovinare", ma deve fare i conti con un fenomeno già notato in Scott (1955) e Artstein & Poesio (2008): dati due diversi schemi di codifica per lo stesso task, quello con il minor numero di categorie registrerebbe una più alta percentuale di I.A.A. Il valore è fortemente influenzato anche dal problema della "prevalenza", ovvero la maggior concentrazione di item in una delle categorie: come avremo modo di discutere in § 2.2.1, una simile distribuzione influenza in negativo la possibilità di raggiungere alti livelli di I.A.A., indipendentemente dalla grandezza del campione.

2.2 Misure "kappa"

Il livello di I.A.A. nell'espressione di giudizi categoriali deve perciò necessariamente essere esplicitato nei termini di eccedenza rispetto all'accordo ottenibile casualmente, pena la mancanza di effettiva informatività. In ambito psicometrico sono stati introdotti numerosi coefficienti statistici in grado di correggere tale aspetto: questi indici, a cui si farà riferimento con il nome di "**misure kappa**", si fondano su tre assunti (Soeken & Prescott, 1986):

- gli *item* soggetti a valutazione sono indipendenti l'uno dall'altro;
- i *rater* che giudicano gli item operano in autonomia ed in modo completamente indipendente;
- le categorie usate sono mutualmente esclusive ed esaustive.

2.2.1 2 rater

Il caso base è rappresentato dai coefficienti per la valutazione dei giudizi prodotti da due soli *rater*, indice noto ai più come "**k di Cohen**". Prima di passare alla presentazione della misura è però necessaria una piccola premessa terminologica. Il celebre articolo di Carletta (1996), a cui va il merito di aver stabilito la valutazione dell'*agreement* come standard *de facto* in LC, ha introdotto una piccola inconsistenza in letteratura (Artstein & Poesio, 2008): la studiosa, nel suggerire l'utilizzo di un coefficiente definito "kappa", fa infatti riferimento non all'originale k proposta in Cohen (1960), ma ad una misura molto simile, introdotta cinque anni prima da Scott. La questione non si esaurisce in un mero problema terminologico: esistono infatti tre indici che, pur condividendo la medesima formula, sono fondati su ipotesi diverse riguardo la distribuzione degli item nelle categorie, ovvero **S di Bennett *et al.***, **π di Scott** e **k di Cohen**. Le differenti ipotesi soggiacenti comportano diverse modalità di calcolo e quindi risultati non coincidenti, seppure in misura minima. La formula di base è la seguente:

1) $S, \pi, k = \frac{A_0 - A_e}{1 - A_e}$

dove A_e è l'*agreement* dovuto al caso ("*Expected Agreement by chance*"); $A_o - A_e$ stima perciò l'*agreement* effettivamente raggiunto al di sopra della soglia della casualità, mentre $1 - A_e$ misura quanto accordo eccedente il caso è ottenibile. Mentre A_o è estremamente semplice da calcolare (§ 2.1) e ha lo stesso valore nelle tre misure, A_e richiede invece un modello del comportamento degli annotatori. Tutti i coefficienti assumono l'indipendenza dei due annotatori che valutano gli item: la probabilità che due *rater* (r_1 ed r_2) siano d'accordo su una determinata categoria *c* è dunque data dal prodotto della probabilità che ciascun *rater* assegni un item a quella categoria, ovvero:

2) $P(c|r1) \cdot P(c|r2)$

A_e è dato dalla sommatoria di tale probabilità congiunta per tutte le categorie dello schema di codifica.

3) $A_e^S = A_e^\pi = A_e^k = \sum_{c \in C} P(c|r_1) \cdot P(c|r_2)$

La differenza tra S, π e k risiede negli assunti che sono alla base del calcolo di $P(c|r_i)$.

S (Bennett *et al.*, 1954) assume che un'annotazione totalmente casuale determini una distribuzione uniforme degli item nelle categorie, ovvero che tutte le categorie dello schema di codifica siano ugualmente probabili; la probabilità che ogni *rater* assegni un item alla categoria *c* è dunque $1/c$.

4) $A_e^S = \sum_{c \in C} \frac{1}{c} \cdot \frac{1}{c} = c \cdot \left(\frac{1}{c}\right)^2 = \frac{1}{c}$

Nell'esempio di tab.1 A_e^S=0.333 e S=0.775.

L'assunto dell'uniformità è un prerequisito estremamente vincolante: per tale ragione non risultano, ad oggi, studi di I.A.A. in LC in cui sia stato impiegato questo coefficiente. In aggiunta, come è stato notato da Scott (1955: 322-323) e riportato da Artstein & Poesio (2008: 561), il valore dell'indice può essere aumentato semplicemente inserendo nello schema di codifica categorie vuote.

Il coefficiente π (Scott, 1955), noto anche col nome di K di Siegel & Castellan (1988), assume che se l'attribuzione degli item alle categorie avviene in modo casuale, la distribuzione sarà uguale per entrambi gli annotatori. $P(c|r_i)$ corrisponderà perciò al rapporto tra il numero totale di assegnazioni alla categoria *c* da parte di entrambi i rater , n_c , e il numero totale di assegnazioni compiute, $2i$.

5) $A_e^\pi = \sum_{c \in C} \left(\frac{n_c}{2i}\right)^2$

Nel caso in oggetto, A_e^π= 0.414 e π=0.744.

k (Cohen, 1960) prevede infine una distribuzione degli item nelle categorie distinta ed unica per ciascun annotatore, rappresentata nelle frequenze marginali della tabella di contingenza.

6) $P(c|r_i) = \frac{n_{r_i c}}{i}$

7) $A_e^k = \sum_{c \in C} \frac{n_{r_1 c}}{i} \cdot \frac{n_{r_2 c}}{i}$

Nell'esempio oggetto di discussione, pertanto, A_e^k=0.41 e k=0.764.

La corretta scelta dell'indice non può prescindere dalla considerazione che i coefficienti sono fortemente influenzati da disomogeneità nella distribuzione dei dati (Feinstein & Cicchetti, 1990; Cicchetti & Feinstein 1990; Di Eugenio & Glass, 2004; Artstein & Poesio, 2008), classificabili in due tipologie principali: la già ricordata "prevalenza" (tab. 2) e il "*bias*", cioè il grado con cui gli annotatori sono in accordo/disaccordo nelle loro valutazioni complessive, ossia le loro "tendenze" nell'esprimere giudizi (tab. 3 e 4).

		rater 1			
		*c*1	*c*2	*c*3	tot
rater 2	*c*1	18	0	1	19
	*c*2	0	0	0	0
	*c*3	1	0	0	1
	tot	19	0	1	20

Tab. 2: Distribuzione affetta da prevalenza.

		rater 1			
		*c*1	*c*2	*c*3	tot
rater 2	*c*1	4	1	1	6
	*c*2	1	3	3	7
	*c*3	1	2	4	7
	tot	6	6	8	20

Tab. 3: Distribuzioni marginali simili.

		rater 1			
		*c*1	*c*2	*c*3	tot
rater 2	*c*1	4	3	1	8
	*c*2	0	3	0	3
	*c*3	1	4	4	9
	tot	5	10	5	20

Tab. 4: Esempio di *bias*, evidente dalle distribuzioni marginali dissimili ("*skewed*").

Nell'esempio di tab. 2, la forte prevalenza in favore della categoria *c1* fa sì che $A_e^{\pi} = A_e^{k} = 0.905$. Di conseguenza, nonostante A_o sia molto alto (0.9), $\pi = k = -0.053$, al di sotto della soglia della pura casualità.

Si confrontino quindi i dati delle tabelle 3 e 4: sebbene entrambe registrino un A_o di 0.55, nel caso in cui le distribuzioni marginali siano molto simili (tab.3) $A_e^{\pi} = 0.335$, $A_e^{k} = 0.336$, $\pi = 0.322$, $k = 0.323$; l'effetto di *bias* (tab.4), invece, affligge la k di Cohen, in ragione delle modalità di calcolo di $P(c|r_1)$: $A_e^{\pi} = 0.334$, $A_e^{k} = 0.287$, $\pi = 0.326$, $k = 0.368$. La differenza tra π e k è empiricamente minima: $A_e^{\pi} \geq A_e^{k}$, perciò $\pi \leq k$. I due coefficienti assumono lo stesso valore nel caso (limite) in cui le distribuzioni marginali dei due *rater* siano identiche, come in tab. 2.

A fronte di ciò, laddove non sia possibile effettuare uno studio che coinvolga più di due *rater*, sembrerebbe pertanto da preferire il coefficiente π di Scott, in grado di generalizzare il comportamento dei singoli annotatori. In letteratura sono state fatte varie proposte riguardo la modalità di presentazione dei risultati dell'I.A.A per due annotatori: allo stato dell'arte sembrerebbe preferibile adottare la soluzione suggerita da Byrt *et al.* (1993) e adottata da Di Eugenio & Glass (2004), ovvero presentare congiuntamente diversi coefficienti:

- k, che in linea di principio meglio si adatta alla valutazione di annotazioni che coinvolgono dati linguistici, e rende conto di eventuali tendenze dei *rater*;
- π, immune all'effetto di *bias*;
- una terza misura, $2A_o$-1, in grado di neutralizzare l'effetto di prevalenza (Byrt *et al.*, 1993).

2.2.2 Possibili estensioni

Sono state proposte moltissime generalizzazioni dei coefficienti presentati, per assicurare maggiore flessibilità ed adattabilità agli specifici task:[1] tra le più note vi è la "***weighted kappa***" (Cohen, 1968), $\mathbf{k_{(w)}}$, indice che consente di esprimere delle gradazioni di disaccordo mediante una tabella di "pesi" di valore compresi tra 0 e 1 ("*weighting scheme*"), come nell'esempio:

	*c*1	*c*2	*c*3
*c*1	1	0	0.5
*c*2	0	1	0.5
*c*3	0.5	0.5	1

Tab.4: Esempio di *weighting scheme*

$A_{o(w)}$ e $A_{e(w)}$ vengono calcolati in modo affine alla k di Cohen (1960), moltiplicando però, in aggiunta, ogni cella della tabella di contingenza per il corrispettivo peso.

$$8) \quad k_{(w)} = \frac{A_{0(w)} - A_{e(w)}}{1 - A_{e(w)}}$$

Se applicata ai dati di Tab.1, $k_{(w)} = 0.774$.

Sono stati inoltre introdotti indici in grado di quantificare l'I.A.A. tra tre o più annotatori: in primis la cosiddetta k di Fleiss (1971), che estende l'indice π di Scott ("***multi-π***"), ed il coefficiente presentato in Davies & Fleiss (1982) che generalizza la k di Cohen ("***multi-k***");[2] ma soprattutto il coefficiente **α di Krippendorff** (1980), che esprime l'I.A.A. in termini di *disagreement*, osservato (D_o) e dovuto al caso (D_e):

$$9) \qquad \alpha = 1 - \frac{D_o}{D_e}$$

La formula, pur essendo stata derivata dalla misura della varianza, non fa esplicito riferimento alle medie dei campioni e può pertanto essere generalizzata ad una moltitudine di schemi di codifica in cui le categorie non siano interpretabili come valori numerici; come per la *weighted kappa* si possono inoltre attribuire pesi alle di-

[1] Alcune estensioni delle misure "kappa", troppo complesse per essere descritte esaurientemente in questa sede, consentono ad esempio di valutare l'I.A.A nel caso in cui i *rater* effettuino osservazioni multiple, e non necessariamente di ugual numero, oppure di gestire gli schemi di annotazione che prevedono la possibilità di attribuire più di una classificazione agli item (Kraemer, 1980).

[2] Le modalità di calcolo sono affini ai coefficienti già descritti. Per i dettagli si rinvia perciò a Fleiss (1971), Davies & Fleiss (1982) e all'ottima sintesi di Artstein & Poesio (2008) e Artstein (2017). Si noti che A_o non potrà essere definito come "percentuale di item su cui c'è accordo", visto che con altissima probabilità ci saranno nei dati item su cui alcuni *rater* saranno d'accordo e altri no: la soluzione proposta in letteratura a partire da Fleiss (1971) è di misurare l'I.A.A. "*pairwise*", ovvero "a coppie".

verse tipologie di *disagreement*, utilizzando *weighting scheme* oppure introducendo nel calcolo delle metriche, ad esempio l'indice statistico MASI (Passonneau, 2006; Dorr *et al.*, 2010).[3] α è equivalente a multi-π per campioni numerosi, ma è in grado, non imponendo un numero minimo di *item*, di mitigare gli effetti statistici di *dataset* a bassa numerosità campionaria; inoltre, consentendo la gestione di *dataset* incompleti, è utilizzabile (o addirittura preferibile) nel caso in cui l'annotazione si svolga in maniera collaborativa e distribuita, ad esempio su piattaforme di *crowdsourcing*.

3 *Reliability*: *agreement* o correlazione?

In letteratura, in particolare in ambito clinico (Bishop & Baird, 2001; Van Noord & Prevatt, 2002; Massa *et al.*, 2008; Gudmundsson & Gretarsson, 2009), non è infrequente che, nella stima dell'I.A.A, vengano preferiti o affiancati alle misure presentate la statistica χ^2 oppure gli indici statistici di correlazione (coefficiente R di Pearson *in primis*, ma anche i non parametrici ρ di Spearman e τ di Kendall).
Come già notato da Cohen (1960), l'utilizzo del χ^2 è una prassi da considerarsi scorretta, poiché la statistica, applicata alla tavola di contingenza, misura casualità e grado di associazione tra i set di giudizi, non l'*agreement* (Banerjee *et al.*, 1999).

> *"[...] Many investigators have computed χ^2 over the table for use as a test of the hypothesis of chance agreement, and some have gone on to compute the contingency coefficient (C) as a measure of degree of agreement. [...] It is readily demonstrable that the use of χ^2 (and therefore the C which is based on it) for the evaluation of agreement is indefensible. When applied to a contingency table, χ^2 tests the null hypothesis with regard to association, not agreement.*
>
> (Cohen, 1960: 38)

Altrettanto scorretta dal punto di vista metodologico è l'applicazione di coefficienti di correlazione inter-/intra- classe, che ugualmente non quantificano l'I.A.A. ma la forza di associazione tra gruppi di valori (Bland & Altman, 1986; Kottner *et al.*, 2011; Stolarova *et al.*, 2014). Si noti inoltre che, dal punto di vista empirico, un'ottima correlazione tra annotazioni può essere raggiunta anche in caso di completa mancanza di accordo, se due set di giudizi differiscono sistematicamente.
La ragione di tali fraintendimenti deve probabilmente essere rintracciata nell'uso sostanzialmente sinonimico dei termini "*reliability*" e "*agreement*" (Stemler, 2004); come puntualizzato da Krippendorff (2004), in realtà:

> *"To be clear, agreement is what we measure; reliability is what we wish to infer from it."*
>
> (Krippendorff, 2004: 413)

Le correlazioni statistiche possono senza dubbio costituire un'informazione interessante nella valutazione globale dell'affidabilità di un *dataset*, a patto però che tale nozione sia tenuta distinta dall'I.A.A. in senso stretto.

4 La valutazione dei coefficienti

La valutazione dei valori assunti dai coefficienti *chance-corrected* rappresenta, ad oggi, un aspetto critico: gli indici possono assumere valori compresi tra -1 e 1, dove k = 1 corrisponde ad un I.A.A. perfetto, k = 0 ad un I.A.A. completamente casuale e k = -1 ad un perfetto disaccordo. Non è però soddisfacente sapere che k abbia un valore superiore alla totale casualità, ma occorre assicurarsi, piuttosto, che gli annotatori non si discostino troppo dall'*agreement* assoluto (Cohen, 1960; Krippendorff, 1980).
A prescindere dal mero valore numerico, va rilevato come i vari studiosi che hanno tentato di indicare delle soglie di riferimento abbiano sottolineato l'arbitrarietà delle loro proposte: in primis Landis & Koch (1977), a cui si deve la più nota griglia per l'interpretazione dei coefficienti:

Kappa Statistic	Strength of Agreement
< 0.0	Poor
0.00 - 0.20	Slight
0.21 - 0.40	Fair
0.41 – 0.60	Moderate
0.61 – 0.80	Substantial
0.81- 1.00	Almost Perfect

Tab. 5: Griglia per l'interpretazione delle misure k (Landis & Koch, 1977).

Così anche Krippendorff, la cui proposta di rifiutare valori di k inferiori a 0.67, accettare quelli superiori a 0.8 e considerare incerti quelli compresi nel *range* costituisce uno dei principali punti di riferimento in letteratura sull'argomento.

> *"Except for perfect agreement, there are no magical numbers, however."*
>
> (Krippendorff, 2004: 324)

Va infine rilevato come il *disagreement* non sia necessariamente indice di bassa qualità

[3] MASI è basato sul coefficiente di Jaccard (1908) e quindi stabilisce la somiglianza/diversità tra insiemi campionari in termini di distanza.

dell'annotazione, scarso *training* degli annotatori o di *guideline* mal definite (Aroyo & Welty, 2015), soprattutto nei task di natura semantica; ed anche che, per aumentare l'affidabilità del *dataset* annotato, non debba necessariamente essere evitato o eliminato: in LC la sua presenza può infatti essere sfruttata esplicitamente, per migliorare le performance di sistemi automatici (come ad esempio in Chklovski & Mihalcea, 2003; Plank, Hovy & Søgaard, 2014).

5 Conclusioni

Come suggerito nei paragrafi iniziali, un alto livello di I.A.A. non costituisce un risultato in sé, ma soltanto uno fra gli indicatori della reale affidabilità dell'annotazione sottoposta a validazione. È perciò auspicabile che un sempre maggior numero di dati sull'I.A.A. nei diversi task di annotazione sia condiviso dai ricercatori, in modo da facilitare l'emergere per confronto dei valori di riferimento.

Bibliografia

Aroyo, L. & Welty, C. (2015). Truth Is a Lie: Crowd Truth and the Seven Myths of Human Annotation. *AI Magazine,* 36 (1):15–24.

Artstein, R. (2017). Inter-annotator Agreement. In: Ide, N. & Pustejovsky, J. (eds.), *Handbook of Linguistic Annotation*. Springer, Dordrecht, pp. 297–314.

Artstein, R. & Poesio, M. (2008). Inter-Coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596.

Bayerl, P. S. & Paul, K. I. (2011). What determines Inter-Coder Agreement in manual annotations? A meta-analytic investigation. *Computational Linguistics*, 37(4):699–725.

Banerjee, M., Capozzoli, M., McSweeney, L. & Sinha, D. (1999). Beyond Kappa: A Review of Interrater Agreement Measures. *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 27(1):3–23.

Bennett, Alpert, R. & Goldstein, A. (1954). Communications through limited response questioning. *Public Opinion Quarterly*, 18:303–308.

Bishop, D.V. & Baird, G. (2001). Parent and teacher report of pragmatic aspects of communication: use of the children's communication checklist in a clinical setting. *Developmental medicine and child neurology*, 43:809–818.

Bland, M. J. & Altman, D. (1986). Statistical methods for assessing agreement between two methods of clinical measurement. *Lancet*, 327:307–310.

Byrt, T., Bishop, J. & Carlin, J. B. (1993). Bias, prevalence and kappa. *Journal of Clinical Epidemiology*, 46(5):423–9.

Carletta, J. (1996). Assessing agreement on classification tasks: the kappa statistic. *Computational Linguistics*, 22(2):249–254.

Chklovski, T. & Mihalcea, R. (2003). Exploiting Agreement and Disagreement of Human Annotators for Word Sense Disambiguation. In: *Proceedings of the Conference on Recent Advances in Natural Language Processing* (RANLP 2003).

Cicchetti, D.V. & Feinstein, A.R. (1990). High Agreement but low Kappa: II. Resolving the paradoxes. *Journal of Clinical Epidemiology*, 43:551–558.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37.

Cohen, J. (1968). Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin*, 70(4):213–220.

Davies, M. & Fleiss, J. L. (1982). Measuring Agreement for Multinomial Data. *Biometrics*, 38(4):1047–1051.

Di Eugenio, B. (2000). On the usage of Kappa to evaluate agreement on coding tasks. In: Calzolari N. *et al.* (eds): *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC 2000)*, ELRA - European Language Resources Association, Paris, pp. 441–444.

Di Eugenio, B. & Glass, M. (2004). The Kappa statistic: a second look. *Computational Linguistics*, 30(1):95–101.

Dorr, B.J., Passonneau, R.J., Farwell, D., Green, R., Habash, N., Helmreich, S., Hovy, E., Levin, L., Miller, K. J., Mitamura, T., Rambow, O. & Siddharthan, A. (2010). Interlingual annotation of parallel text corpora: A new framework for annotation and evaluation. *Journal of Natural Language Engineering*, 16(3):197–243.

Feinstein, A.R. & Cicchetti, D.V. (1990). High agreement but low kappa: I. The problems of two paradoxes. *Journal of Clinical Epidemiology*, 43:543–549.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Goodman, L. A. & Kruskal, W. H. (1954). Measures of association for cross classifications. *Journal of*

the American Statistical Association, 49(268): 732–764.

Gudmundsson, E. & Gretarsson, S. J. (2009). Comparison of mothers' and fathers' ratings of their children's verbal and motor development. *Nordic Psycholgy*, 61:14–25.

Jaccard, P. (1908). Nouvelles recherches sur la distribution florale. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 44:223–270.

Kottner, J., Audige, L., Brorson, S., Donner, A., Gajewski, B.J., Hróbjartsson, A., Roberts, C., Shoukri, M. & Streiner, D.L. (2011). Guidelines for reporting reliability and agreement studies (GRRAS) were proposed. *International journal of nursing studies*, 48:661–671.

Kraemer, H. C. (1980). Extension of the kappa coefficient. *Biometrics*, 36(2):207–16.

Krippendorff, K. (1980). *Content Analysis: an introduction to its Methodology*. Sage Publications, Thousand Oaks, CA.

Krippendorff, K. (2004). Reliability in content analysis: Some common misconceptions and recommendations. *Human Communication Research*, 30(3):411–433.

Landis, J. R. & Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174.

Massa, J., Gomes, H., Tartter, V., Wolfson, V. & Halperin, J.M. (2008). Concordance rates between parent and teacher clinical evaluation of language fundamentals observational rating scale. *International Journal of Language & Communication Disorders*, 43:99–110.

Mathet, J., Widlöcher, A. & Métivier, J. (2015). The Unified and Holistic Method Gamma (γ) for Inter-Annotator Agreement Measure and Alignment. *Computational Linguistics*, 41(3):437–479.

Passonneau, R. (2006). Measuring agreement on set-valued items (MASI) for semantic and pragmatic annotation. In: *Proceedings of the fifth International Conference on Language Resources and Evaluation (LREC 2006)*. ELRA European Language Resources Association, Paris.

Plank, B., Hovy, D. & Søgaard, A. (2014). 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*, pp. 742–751.

Scott, W.A. (1955). Reliability of Content Analysis: The Case of Nominal Scale Coding. *Public Opinion Quarterly*, 19(3):321–325.

Siegel, S. & Castellan, J. (1988). *Nonparametric statistics for the behavioral sciences*. McGraw-Hill, Boston, MA.

Soeken, K.L. & Prescott, P.A. (1986). Issues in the use of kappa to estimate reliability. *Medical Care*, 24(8):733–41.

Stemler, S.E. (2004). A Comparison of Consensus, Consistency, and Measurement Approaches to Estimating Interrater Reliability. *Practical Assessment, Research & Evaluation*, 9:66–78.

Stolarova, M., Wolf, C., Rinker, T. & Brielmann, A. (2014). How to assess and compare inter-rater reliability, agreement and correlation of ratings: an exemplary analysis of mother-father and parent-teacher expressive vocabulary rating pairs. *Frontiers in psychology*, 5, 509.

Van Noord, R.G. & Prevatt, F.F. (2002). Rater agreement on IQ and achievement tests: effect on evaluations of learning disabilities. *Journal of School Psychology,* 40(2):167–176.

Warrens, M. J. (2010). Inequalities between multi-rater kappas. *Advances in Data Analysis and Classification*, 4(4):271–286.

Auxiliary selection in Italian intransitive verbs: a computational investigation based on annotated corpora

Ilaria Ghezzi
Dipartimento di Lingue e Letterature Straniere e Culture Moderne
Università degli Studi di Torino
ghezzi.ila@gmail.com

Cristina Bosco
Alessandro Mazzei
Dipartimento di Informatica
Università degli Studi di Torino
{bosco,mazzei}@di.unito.it

Abstract

English. The purpose of this paper is the analysis of the auxiliary selection in intransitive verbs in Italian. The applied methodology consists in comparing the linguistic theory with the data extracted from two different annotated corpora: UD-IT and PoSTWITA-UD. The analyzed verbs have been classified in different semantic categories depending on the linguistic theory. The results confirm the theoretical assumptions and they could be considered as a starting point for many applicative tasks as Natural Language Generation.

Italiano. *Obiettivo di questo lavoro è l'analisi della selezione dell'ausiliare dei verbi intransitivi in italiano. La metodologia applicata consiste nel confrontare la teoria linguistica con dati estratti da due corpora annotati: UD-IT e PoSTWITA-UD. I verbi analizzati sono stati classificati nelle categorie semantiche individuate partendo dalla letteratura teorica. I risultati confermano con buona approssimazione gli assunti teorici e possono quindi essere il punto di partenza per l'implementazione di strumenti come sistemi di Natural Language Generation.*

1 Introduction

In this work we have applied a corpus-based approach to the investigation of the behavior of Italian intransitive verbs for what concerns the selection of the auxiliary verb. We considered two corpora, namely UD-IT[1] and PoSTWITA-UD (Sanguinetti et al., 2018), annotated following the Universal Dependencies standards. UD-IT and PoSTWITA-UD are treebanks (morphologically and syntactically annotated corpora) for the Italian language. UD-IT is made up of texts from various sources, namely the Italian Constitution, the Italian Civil Code, newspaper articles and Wikipedia. It is a balanced corpus and, therefore, a representative corpus for Italian standard language. On the other hand, PoSTWITA-UD contains tweets from the social media Twitter, and can therefore be considered a representative corpus for the Italian Language used in social media (non-standard Italian). This difference allows us to investigate verbs' behaviour in standard and non-standard Italian Language.

Intransitive verbs have been extensively studied in both traditional grammar and linguistics, since they do not always follow a standardized rule for the auxiliary selection (see examples Section 2). This fact could be the reason why their status is not currently formalized enough in NLP, as long as Italian is concerned. Among the most recent investigation which use a corpus linguistic methodology for the Italian language, we find (Amore, 2017).

Our analysis starts from traditional Italian grammars and then moves to the Auxiliary Selection Hierarchy by (Sorace, 2000), a syntactic and semantic perspective on the behaviour of intransitive verbs and auxiliary selection in Romance languages. That can be useful for formalizing the studied phenomenon and thus providing Natural Language Generation systems with the necessary information regarding the auxiliary selection, which is our final goal. Another contribute for the same systems but for what concerns adjectives has been published in (Conte et al., 2017).

2 Auxiliary Selection in Italian

As in several other languages, in Italian one among two auxiliary verbs can be used together

[1] http://universaldependencies.org/it/overview/introduction.html

with the past participle verbal forms for compounding periphrastic tenses: *avere* (to have) and *essere* (to be), henceforth respectively indicated as A or E. When the verb is transitive, the auxiliary selection follows standard rules, depending on the diathesis: transitive verbs in active diathesis select A (e.g. *Luca ha mangiato la mela* – Luca ate the apple) while transitive verbs in passive diathesis select E (e.g. *La mela è mangiata da Luca* – The apple is eaten by Luca).

Problems in the auxiliary selection occur instead when the verb is intransitive. In fact, provided that the behaviour of intransitive verbs depends on both semantic and syntactic factors (Van Valin, 1990), a general rule for their auxiliary selection cannot always be formulated[2] (Patota, 2003). Some intransitive verbs can actually select both A or E depending on the semantics of the sentence, while others only admit E or A. See the examples[3] below:

1. *Maria ha corso alle olimpiadi / Maria è corsa a casa*
   (Maria has run at the Olympics / Maria is run home)

2. *Ieri ho camminato al parco / *Ieri sono camminato al parco*[4]
   (I walked in the park yesterday)

Even if all the verbs involved describe a form of movement and are semantically similar, in the first couple of examples the intransitive verb *correre* (to run) allows the selection of both E and A, while in the second one the intransitive verb *camminare* (to walk) only allows the selection of A, and the sentence generated by selecting E is indeed ungrammatical.

Traditional and normative Italian grammars do not provide an analysis of intransitive verbs and auxiliary selection which could be formalized and therefore usefully spent in NLP. In fact, they only suggest lists of verbs that select A or E as auxiliary, see e.g. (Moretti and Orvieto, 1979), (Patota, 2003), (Renzi et al., 1991), (Serianni, 1988), (Dardano and Trifone, 1997). For this reason, we decided to consider other theories too, starting from the *Unaccusative Hypothesis* discussed in (Perlmutter, 1978) and moving to the *Auxiliary Selection Hierarchy* proposed in (Sorace, 2000).
Moreover, we considered the application of a corpus-based approach, provided that corpora represent the way Italian native speakers use A or E together with intransitive verbs. We hypothesized that, this kind of probabilistic perspective can allow a reliable description of the phenomenon. In fact, when there is a lack of standard grammar rules, it is possible to determine certain linguistic aspects by extracting data from corpora. Doing so, we can compensate the lack of standard grammar rules with probabilistic and statistic data.

2.1 The theoretical status of intransitive verbs

For accounting for the behavior of intransitive verbs, in 1978, Perlmutter expressed the *Unaccusative Hypothesis*, which splits intransitive verbs in 2 subcategories: the *unaccusative verbs* and the *unergative verbs*. Perlmutter suggested that the unaccusative verbs are intransitive verbs whose grammatical subject is not an agent (e.g. *La nave è affondata* – The ship is sunk), while unergative verbs are intransitive verbs whose grammatical subject is an agent (e.g. *Giulia ha camminato* - Giulia has walked).

More recently other linguists and researchers analysed the topic, following two major lines: Rosen that suggested to follow a syntactic-only approach (Rosen, 1984), Van Valin and Dowty that suggested a semantic-only approach (Van Valin, 1990; Dowty, 1979).
A development of Perlmutter's hypothesis supported by experimental and psycho-linguistic results can be found in Sorace (2000) that proposed an interesting modelling of the behaviour of intransitive verbs with respect to the selection of auxiliary for Italian too. This theory especially inspired our current work.

2.2 A hierarchy for auxiliary selection

According to the theory proposed by Sorace, intransitive verbs can be hierarchically organized according to their different degree of telicity and agentivity. The more a verb is telic or agentive, the more it systematically selects the auxiliary verb E or A respectively.
This hierarchy of intransitive verbs, also known as *Auxiliary Selection Hierarchy* (ASH), includes categories defined on the basis of thematic and as-

[2]Flexibility in auxiliary selection can be accounted for a large number of cases if context is taken into account.

[3]The translation of the examples can be not correctly mapped on the English rules. When this happens the auxiliary is underlined.

[4]Sentences marked with * are ungrammatical.

ASH category	examples	auxiliary selection
Change of location (**maximum telicity**)	to go, to arrive	selects E
Change of state	to appear, to happen	
Continuation of pre-existing state	to stay, to last	
Existence of state	to exist, to seem	
Uncontrolled process	to sleep, to rain	
Controlled process - motional	to walk, to run	
Controlled process - non motional (**maximum agentivity**)	to act, to play	selects A

Table 1: Examples of verbs organized in the ASH: at the poles verbs that always select E or always select A, and between the verbs that alternatively select both.

pectual features. At one end of the ASH we find intransitive verbs which categorically select E as auxiliary, while at the other end we find intransitive verbs that always select A. The verbs between the two poles of the ASH can have an alternation in the auxiliary selection.
The ASH has been exploited in our work for classifying Italian intransitive verbs depending on its categories which are reported and exemplified in Table 1. This classification may seem wrong for verbs like "to go" (*andare*), which are both agentive and unaccusative, but, as Sorace (2000:863) points out, the verbs that express a change of location have the highest degree of dinamicity and telicity, and they always select E as auxiliary.

3 Intransitive verbs in the fundamental Italian vocabulary

3.1 Verbs selection

In order to focus our study on the intransitive verbs that are more commonly and competently used by Italian speakers, we decided to extract the intransitive verbs to be studied from the *Nuovo vocabolario di base della lingua italiana* (Chiari and De Mauro, 2016), a well known reference resource for Italian lexicography. The lexical entries are here organized in three basic vocabulary ranges according to their frequency of use and ease of recovery in speakers' brain: fundamental vocabulary (FO), high usage (AU) and high availability (AD).
For the present work, we considered only the verbs of the FO vocabulary, for a total of 51 intransitive verbs. But some of these verbs showed more than one single meaning and they could therefore be included in different categories of Sorace's ASH. In order to carry out a disambiguation process, we used Babelnet[5], a multilingual lexicalized semantic network and ontology. After the disambiguation process, the total number of verbs is 67.
For what concerns intransitive pronominal verbs (e.g.*rompersi, "to break"*), we decided not to take them into consideration for our research, since they always select the auxiliary E when constructed in compound tenses (eg. *Gli occhiali si sono rotti* (The glasses broke)). The choice to limit our research to the FO vocabulary is due to the fact that one should expect an expert usage of the verbs of this class also by an artificial speaker.

3.2 Verbs classification

After having selected the verbs, we proceeded to their classification, following the theory proposed by (Sorace, 2000). The intransitive verbs belonging to the FO Italian vocabulary have therefore been included in different categories, depending both on the semantics and the syntax.

Table 2 shows some examples of Italian intransitive verbs belonging to the FO class, classified depending on the ASH by Sorace (2000).

ASH	FO verbs
Change of location	andare (to go)
Change of state	apparire (to appear)
Contin. pre-existing state	rimanere (to last)
Existence of state	esistere (to exist)
Uncontrolled process	dormire (to sleep)
Control. proc. (motion)	camminare (to walk)
Control. proc. (nonmotion)	agire (to act)

Table 2: Examples of intransitive verbs belonging to FO and classified according to ASH.

[5]https://babelnet.org/

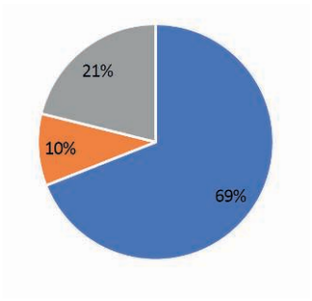


Figure 1: The percentage of intransitive verbs selecting E (in blue), A (in orange) or not detected (in grey) in UD-it.

4 Reference corpora

As mentioned above, the reference corpora for this work are the treebanks UD-IT and PoSTWITA-UD, both annotated according to the Universal Dependencies (UD) format for what concerns morphology and syntax. Provided that UD is currently a standard de facto, the exploitation of this format allows us the application of the same methodology on other resources or languages.

The exploitation of both the data set is motivated by the need to extend our research on the larger available amount of data, and by the fact that UD-IT is representative of the standard Italian language, while PoSTWITA-UD represents the Italian language used in social media. This allows us to obtain a comprehensive set of results.

4.1 Data extraction

To extract the data concerning the auxiliary selection on UD-it and PoSWITA we used the Sets Treebank Search provided by the University of Turku, available for free at `http://bionlp-www.utu.fi/dep_search/`.

We formulated an expression that allowed us to extract data related only to intransitive verbs that appear in the reference corpora at the past participle form together with an auxiliary verb (A or E). We then compared the data from the corpora against the classification based on the linguistic theory.

5 Results

After the data extraction from UD-IT and PoSTWITA-UD, a first consideration is to be made about the percentages of intransitive verbs that select A or E in the two corpora.

As figure 1 shows, in UD-IT the auxiliary A is selected by 10% of the verbs and the auxiliary E by 69%. As long as PoSTWITA-UD is concerned (see fig.2), 49% of verbs select E and 9% select A in this corpus. The remaining percentages (in grey) are made up by the verbs that do not appear in compound tenses in the corpus and did not provide useful result for our study; they must be studied in larger corpora.

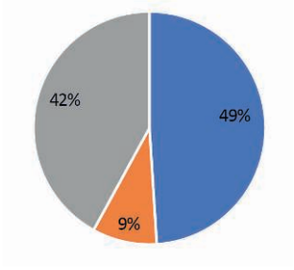


Figure 2: The distribution of verbs selecting E (in blue) and A (in orange) in postwita-UD.

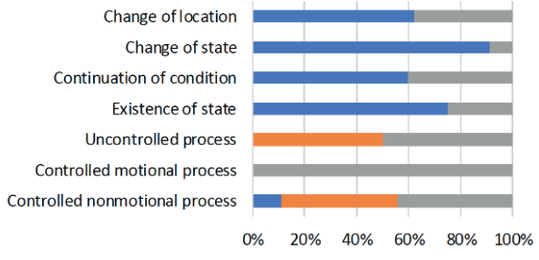


Figure 3: The distribution of verbs selecting E (in blue) and A (in orange) across Sorace's verbal classes in postwita-UD.

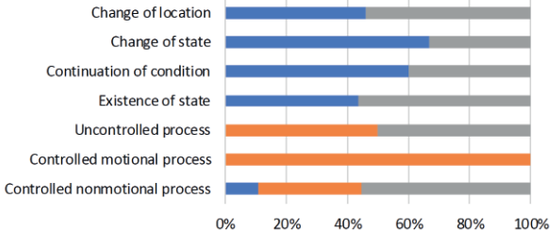


Figure 4: The distribution of verbs selecting E (in blue) and A (in orange) across Sorace's verbal classes in it-UD.

The overall results confirm the linguistic theory for what concerns the distribution in semantic classes organized by Sorace in hierarchy. In fact, as Sorace affirms in (Sorace, 2000), the auxiliary E is selected by intransitive verbs belonging

to the categories of Change of location, Change of state, Continuation of condition and Existence of state as shown in figure 3 and 4 with respect to our two reference corpora. Figure 5 shows an example with the verb "to go" taken from UD-it.

On the other hand, the auxiliary A is selected

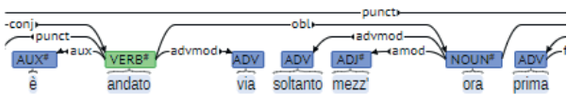


Figure 5: Example taken from UD-IT. In English: "He has gone away only half an hour before the end".

by verbs belonging to the categories of Uncontrolled process, Controlled motional Process and Controlled nonmotional process. This is an example taken from the corpus UD-It, for the verb "to act", *agire* in Italian: *Se, a richiesta del mittente, il vettore emette la lettera di trasporto aereo, si considera, sino a prova contraria, che egli abbia agito in nome del mittente* [6].

As fig. 4 shows, the results related to the category of "controlled nonmotional process" show that both auxiliary A and E can be admitted. This fact is also mentioned by (Sorace, 2000), when she says that some Italian native speakers may accept the auxiliary verb E for this category of verb (e.g. *Il cibo dell'ONU ha / è funzionato solo come palliativo*).

6 Conclusion and future work

The paper presents a study about the auxiliary selection in intransitive verbs in Italian. Providing that the qualitative description given by traditional grammars does not allow the definition of a formal model for the auxiliary selection, we considered a study (Sorace, 2000) that classifies the intransitive verbs taking into account both semantic and syntactic features and behaviors. The long-term goal of this study is to contribute to the development of a natural language generation system for Italian (Mazzei et al., 2016; Mazzei, 2016; Conte et al., 2017). In particular, the facilities of a fluent automatic selection of the auxiliary can be an important feature also in context where the realizer module of the system is used for extracting suggestions for non-native speakers learning Italian as L2.

We adopted in this study a corpus-based perspective and we tested our assumption on two treebanks for Italian respectively representig standard and social media language. The results confirm and validate the theory and they could be used to develop a formal model that can be exploited in a computational context.

[6]English translation: If, under request of the sender, the carrier issues the airway bill, it is considered, if not proven otherwise, that he has acted in the name of the sender.

References

M. Amore. 2017. I verbi neologici nell'italiano del web: Comportamento sintattico e selezione dell'ausiliare. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017), Rome, Italy, December 11-13, 2017.*

I. Chiari and T. De Mauro. 2016. *Nuovo vocabolario di base della lingua italiana*.

G. Conte, C. Bosco, and A. Mazzei. 2017. Dealing with Italian adjectives in noun phrase: a study oriented to natural language generation. In *Proceedings of 4th Italian Conference on Computational Linguistics (Clic-it 2017)*, Rome, Italy.

M. Dardano and P. Trifone. 1997. *La nuova grammatica della lingua italiana*. Zanichelli, Bologna.

D. Dowty. 1979. *Word Meaning and Montague Grammar,*. D. Reidel, Dordrecht.

A. Mazzei, C. Battaglino, and C. Bosco. 2016. SimpleNLG-IT: adapting SimpleNLG to Italian. In *Proceedings of the 9th International Natural Language Generation conference*, pages 184–192, Edinburgh, UK, September 5-8. Association for Computational Linguistics.

Alessandro Mazzei. 2016. Building a computational lexicon by using SQL. 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, pages 1–5. CEUR-WS.org, December.

G.B Moretti and G.R. Orvieto. 1979. *Grammatica italiana*. Benucci, Perugia.

G. Patota. 2003. *Grammatica di riferimento della lingua italiana per stranieri*. Le Monnier, Firenze.

D. M. Perlmutter. 1978. Impersonal passives and the unaccusative hypothesis. In *Proceedings of the Annual Meeting of the Berkeley Linguistics Society 38*. Linguistic Society of America.

L. Renzi, G. Salvi, and A. Cardinaletti. 1991. *Grande grammatica italiana di consultazione*. Il Mulino, Bologna.

C. Rosen. 1984. The interface between semantic roles and initial grammatical relations. In D.M. Perlmutter and C. Rosen, editors, *Studies in Relational Grammar 2*, pages 38–77. University of Chicago Press.

M. Sanguinetti, C. Bosco, A. Lavelli, A. Mazzei, and F. Tamburini. 2018. PoSTWITA-UD: an Italian Twitter treebank in Universal Dependencies. In *Proceedings of 11th International Conference on Language Resources and Evaluation - LREC 2018*, Miyazaki, Japan, 7-12 May.

L. Serianni. 1988. *Grammatica italiana. Italiano comune e lingua letteraria. Suoni, forme e costrutti*. UTET, Torino.

A. Sorace. 2000. Gradients in auxiliary selection with intransitive verbs. *Language*, 76(4):859–890.

R. D. Van Valin. 1990. Semantic parameters of split intransitivity. *Language*, 66(2):221–260.

Constructing an Annotated Resource for Part-Of-Speech Tagging of Mishnaic Hebrew

Emiliano Giovannetti[1], **Davide Albanesi**[1], **Andrea Bellandi**[1], **Simone Marchi**[1], **Alessandra Pecchioli**[2]
[1] Istituto di Linguistica Computazionale, Via G. Moruzzi 1, 56124, Pisa
name.surname@ilc.cnr.it
[2] Progetto Traduzione Talmud Babilonese S.c.a r.l., Lungotevere Sanzio 9, 00153 Roma
alepec3@gmail.com

Abstract

English. This paper introduces the research in Part-Of-Speech tagging of mishnaic Hebrew carried out within the Babylonian Talmud Translation Project. Since no tagged resource was available to train a stochastic POS tagger, a portion of the Mishna of the Babylonian Talmud has been morphologically annotated using an ad hoc developed tool connected with the DB containing the talmudic text being translated. The final aim of this research is to add a linguistic support to the Translation Memory System of Traduco, the Computer-Assisted Translation tool developed and used within the Project.

Italiano. *In questo articolo è introdotta la ricerca nel Part-Of-Speech tagging dell'Ebraico mishnaico condotta nell'ambito del Progetto Traduzione Talmud Babilonese. Data l'indisponibilità di risorse annotate necessarie per l'addestramento di un POS tagger stocastico, una porzione di Mishnà del Talmud Babilonese è stata annotata morfologicamente utilizzando uno strumento sviluppato ad hoc collegato al DB dove risiede il testo talmudico in traduzione. L'obiettivo finale di questa ricerca è lo sviluppo di un supporto linguistico al sistema di Memoria di Traduzione di Traduco, lo strumento di traduzione assistita utilizzato nell'ambito del Progetto.*

1 Introduction

The present work has been conducted within the Babylonian Talmud Translation Project (in Italian, Progetto Traduzione Talmud Babilonese - PTTB) which aims at the translation of the Babylonian Talmud (BT) into Italian.

The translation is being carried out with the aid of tools for text and language processing integrated into an application, called *Traduco* (Bellandi et al., 2016), developed by the Institute of Computational Linguistics "Antonio Zampolli" of the CNR in collaboration with the PTTB team. Traduco is a collaborative computer-assisted translation (CAT) tool conceived to ease the translation, revision and editing of the BT.

The research described here fits exactly in this context: we want to provide the system with additional informative elements as a further aid in the translation of the Talmud. In particular, we intend to linguistically analyze the Talmudic text starting from the automatic attribution of the Part-Of-Speech to words by adopting a stochastic POS tagging approach.

The first difficulty that has emerged regards the text and the languages it contains. In this regard we can say, simplifying, that the Babylonian Talmud is essentially composed of two languages which, in turn, correspond to two distinct texts: the Mishna and the Gemara. The first is the oldest one written in mishnaic Hebrew, one of the most homogeneous and coherent languages appearing in the Talmud that, for this reason, has been chosen to start from in the POS tagging experiment.

The main purpose of linguistic analysis in the context of our translation project is to improve the suggestions provided by the system through the so-called Translation Memory (TM).

Moreover, on a linguistically annotated text it is possible to carry out linguistic-based searches, useful both for the scholar (in this

case a talmudist), and, during the translation work, for the revisor and the curator, who have the possibility, for example, to make bulk editing of polysemous words by discarding out words with undesired POS.

The rest of the paper is organized as follows: Section 2 summarizes the state of the art in NLP of Hebrew. The construction of the linguistically annotated corpus is described in Section 3. The training process and evaluation of the POS taggers used in the experiments is detailed in Section 4. Lastly, Section 5 outlines the next steps of the research.

2 State of the art

The aforementioned linguistic richness and the intrinsic complexity of the Babylonian Talmud make automatic linguistic analysis of the BT particularly hard (Bellandi et al., 2015).

However, some linguistic resources of ancient Hebrew and Aramaic have been (and are being) developed, among which we cite: i) the Hebrew Text Database (Van Peursen and Sikkel, 2014) (ETCBC) accessible by SHEBANQ[1] an online environment for the study of Biblical Hebrew (with emphasis on syntax), developed by the Eep Talstra Centre for Bible and Computer of the Vrije Universiteit in Amsterdam; ii) the Historical Dictionary[2] project of the Academy of the Hebrew Language of Israel; iii) the Comprehensive Aramaic Lexicon (CAL)[3] developed by the Hebrew Union College of Cincinnati; iv) the Digital Mishna[4] project, concerning the creation of a digital scholarly edition of the Mishna conducted by the Maryland Institute of Technology in the Humanities.

Apart from the aforementioned resources, to date there are no available NLP tools suitable for the processing of ancient north-western Semitic languages, such as the different Aramaic idioms and the historical variants of Hebrew attested in the BT. The only existing projects and tools for the processing of Jewish languages (Kamir et al., 2002) (Cohen and Smith, 2007) have been developed for modern Hebrew, a language that has been artificially revitalized from the end of the XIX century and that does not correspond to the idioms recurring in the BT. Among them we cite HebTokenizer[5] for tokenization, MILA (Barhaim et al., 2008), HebMorph[6], MorphTagger [7] and NLPH[8] for morphological analysis and lemmatization, yap[9], hebdepparser[10], UD_Hebrew[11] for syntactic analysis. We conducted some preliminary tests by starting with MILA's (ambiguous) morphological analyzer applied to the three main languages of the Talmud:

1. *Aramaic*: Hebrew and Aramaic are different languages. There are even some cases in which the very same root has different semantics in the two languages. In addition, MILA did not recognize many aramaic roots, tagging the relative words, derived from them, as proper nouns.

2. *Biblical Hebrew*: MILA recognized most of the words, since Modern Hebrew preserved almost the entire biblical lexicon. However, syntax of Modern Hebrew is quite different from that of Biblical Hebrew, leading MILA to output wrong analyses.

3. *Mishnaic Hebrew*: this is the language where MILA performed better. Modern Hebrew inherits some of the morphosyntactic features of mishnaic Hebrew, however, the two idioms differ substantially on the lexicon, since in modern Hebrew many archaic words have been lost (Skolnik and Berenbaum, 2007).

In the light of the above, we decided to create a novel linguistically annotated resource to start developing our own tools for the processing of ancient Jewish languages. In the next section, we will describe how the resource was built.

3 Building the resource

The linguistic annotation of Semitic languages poses several problems. Although we here discuss the analysis of Hebrew, many of the critical points that must be taken into account are

[1] shebanq.ancient-data.org
[2] maagarim.hebrew-academy.org.il
[3] cal.huc.edu
[4] www.digitalmishnah.org
[5] www.cs.bgu.ac.il/∼yoavg/software/hebtokenizer
[6] code972.com/hebmorph
[7] www.cs.technion.ac.il/∼barhaim/MorphTagger
[8] github.com/NLPH/NLPH
[9] github.com/habeanf/yap
[10] tinyurl.com/hebdepparser
[11] github.com/UniversalDependencies/UD_Hebrew

common to other languages belonging to the same family. As already mentioned in the previous section, the first problem concerns the access to existing linguistic resources and analytical tools which, in the case of Hebrew, are available exclusively for the modern language.

One of the major challenges posed by the morphological analysis of Semitic languages is the orthographic disambiguation of words. Since writing is almost exclusively consonantal, every word can have multiple readings. The problem of orthographic ambiguity, crucial in all studies on large corpora (typically in Hebrew and modern Arabic), does not prove to be so difficult when the text under examination is vocalized.

The edition of the Talmud used in the project is actually vocalized and the text, consequently, is orthographically unambiguous. An additional critical aspect is represented by the definition of the tagset. Most of the computational studies on language analysis have been conducted on Indo-european languages (especially on English).

As a result, it may be difficult to reuse tagsets created for these languages. Not surprisingly, there are still many discussions about how it is better to catalog some POS and each language has its own part under discussion. Each tagset must ultimately be created in the light of a specific purpose. For example, the tagging of the (Modern) Hebrew Treebank developed at the Technion (Sima'an et al., 2001) was syntax-oriented, while the work on participles of Hebrew described in (Adler et al., 2008) was more lexicon-oriented. We considered the idea of adopting the tagset used in the already cited Universal Dependency Corpus for Hebrew. However, its 16 tags appeared to be too "coarse grained" for our purposes.[12] In particular, the UD tagset lacks of all the prefix tags that we needed. For this reason we decided to define our own tagset.

Once the tagset has been defined, it remains to decide which is the most suitable grammatical category to associate with each token. You can collect essentially two types of information, the problem is how and if you can keep both, in particular: i) the definition of the token from a syntagmatic perspective (i.e. what the token represents in context) and ii) the lexical information that the token gives by itself (without context). To give a couple of examples:

- Verb/noun: אִשָּׁתּוֹ אֶת הַמַּדִּיר → is הַמַּדִּיר "the one who makes a vow" or "the vowing"? (the one who consecrates his wife): should it be assigned to verb or noun category?
- Adjective/verb: עַד וְלִנְמוֹר לְהַתְחִיל יְכוֹלִין אֵם לַשּׁוּרָה יַגִּיעוּ שֶׁלֹּא - יַתְחִילוּ → is יְכוֹלִין adjective or verb (given that most of the mishnaic language dictionaries provide both options)?

We could discuss about which category would be the best for each and why, but, for now, we decided to keep both by introducing two parallel annotations, by "category" (without context) and by "function" (in context). The tagset we used for this work are the following: *agg.*, *avv.*, *cong.*, *interiez.*, *nome pr.*, *num. card.*, *num. ord.*, *pref. art.*, *pref. cong.*, *pref. prep.*, *pref. pron. rel.*, *prep.*, *pron. dim.*, *pron. indef.*, *pron. interr.*, *pron. pers.*, *pron. suff.*, *punt.*, *sost.*, *vb.*

One could also envisage the refining of the tagset by adding: interrogative, modal, negation, and quantifier (Adler, 2007) (Netzer and Elhadad, 1998) (Netzer et al., 2007).

As anticipated, in order to build the morphologically annotated resource, all of the Mishna sentences were extracted from the Talmud and annotated using an ad hoc developed Web application (Fig. 1).

All the annotations have been made with the aim of training a stochastic POS tagger in charge of the automatic analysis of the entire Mishna: to obtain a good accuracy it was thus necessary to manually annotate as many sentences as possible. To date, 10442 tokens have been annotated.

The software created for the annotation shows, in a tabular form, the information of the analysis carried out on a sentence by sentence basis.

The system, once a sentence is selected for annotation, checks whether the tokens composing it have already been analyzed and, in

[12] github.com/UniversalDependencies/UD_Hebrew-HTB/blob/master/stats.xml

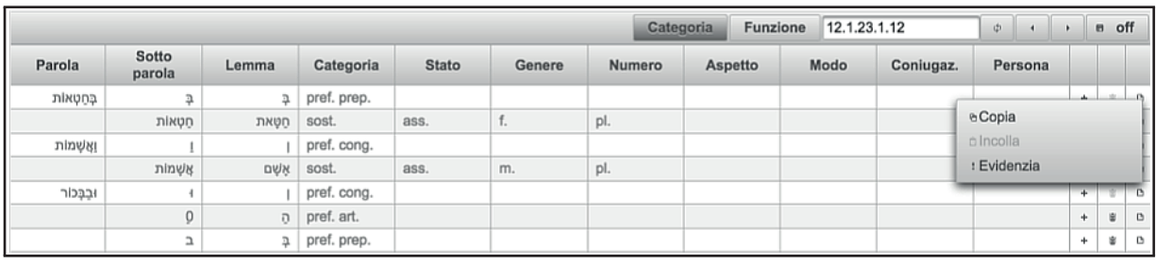


Figure 1: The interface for the linguistic annotation of the corpus to be used to train the POS tagger

case, calculates a possible subdivision into subtokens (i.e. the stems, prefixes and suffixes constituting each word) by exploiting previous annotations. If the system finds that a word is associated with multiple different annotations, it proposes the most frequent one.

Regarding the linguistic annotation, the grammar of Pérez Fernández (Fernández and Elwolde, 1999) was adopted and, for lemmatization, the dictionary of M. Jastrow (Jastrow, 1971).

The software allows to gather as much information as possible for each word by providing a double annotation: by "category" to represent the POS from a grammatical point of view, and by "function" to describe the function the word assumes in its context. For the POS tagging experiments, described below, we used the annotation made by "function".

4 Training and testing of POS taggers

Once the mishnaic corpus has been linguistically annotated three of the most used algorithms for POS tagging have been used and evaluated: HunPos (Halácsy et al., 2007), the Stanford Log-linear Part-Of-Speech Tagger (Toutanova et al., 2003), and TreeTagger (Schmid, 1994). The three algorithms implement supervised stochastic models and, consequently, they need to be trained with a manually annotated corpus.

To evaluate the accuracy of the algorithms we adopted the strategy of *k-fold cross validation* (Brink et al., 2016), with k set to 10, and thus dividing the corpus in 10 partitions.

Table 1 summarizes the results of the experiment by showing the tagging accuracy of the three tested algorithms. With a number of tokens slightly higher than ten thousands the

Tagging Accuracy		
Stanford	*Hunpos*	*Treetagger*
87,90%	86,34%	86,74%

Table 1: Accuracy of the three POS taggers.

Stanford POS tagger provided the best results over HunPos and Treetagger, with an accuracy of 87,9%.

5 Next steps

In this work, the tagging experiments have been limited to the attribution of the Part-Of-Speech: the next, natural step, will be the addition of the lemma. Furthermore, we will try to modify the parameters affecting the behaviour of the three adopted POS taggers (left at their default values for the experiments) and see how they influence the results.

Once the Mishna will be lemmatized, Traduco, the software used to translate the Talmud in Italian, will be able to exploit this additional information mainly to provide translators with translation suggestions based on lemmas, but also to allow users to query the mishnaic text by POS and lemma.

As a further step we will also take into account the linguistic annotation of portions of the Babylonian Talmud written in other languages, starting from the Babylonian Aramaic, the language of the Gemara, which constitutes the earlier portion of the Talmud.

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

Meni Adler, Yael Netzer, Yoav Goldberg, David Gabay, and Michael Elhadad. 2008. Tagging a hebrew corpus: the case of participles. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, and Daniel Tapias, editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Menahem Meni Adler. 2007. *Hebrew Morphological Disambiguation: An Unsupervised Stochastic Word-based Approach*. PhD Thesis, Ben-Gurion University of the Negev.

Roy Bar-haim, Khalil Sima'an, and Yoad Winter. 2008. Part-of-speech Tagging of Modern Hebrew Text. *Nat. Lang. Eng.*, 14(2):223–251, April.

Andrea Bellandi, Alessia Bellusci, and Emiliano Giovannetti. 2015. Computer Assisted Translation of Ancient Texts: the Babylonian Talmud Case Study. In *Natural Language Processing and Cognitive Science, Proceedings 2014*, Berlin/Munich. De Gruyter Saur.

Andrea Bellandi, Davide Albanesi, Giulia Benotto, and Emiliano Giovannetti. 2016. *Il Sistema Traduco nel Progetto Traduzione del Talmud Babilonese*. IJCoL Vol. 2, n. 2, December 2016. Special Issue on "NLP and Digital Humanities". Accademia University Press.

Henrik Brink, Joseph Richards, and Mark Fetherolf. 2016. *Real-World Machine Learning*. Manning Publications Co., Greenwich, CT, USA, 1st edition.

Shay B. Cohen and Noah A. Smith. 2007. Joint Morphological and Syntactic Disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Miguel Pérez Fernández and John F. Elwolde. 1999. *An Introductory Grammar of Rabbinic Hebrew*. Interactive Factory, Leiden, The Netherlands.

Péter Halácsy, András Kornai, and Csaba Oravecz. 2007. HunPos: An Open Source Trigram Tagger. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 209–212, Stroudsburg, PA, USA. Association for Computational Linguistics.

Marcus Jastrow. 1971. *A dictionary of the Targumim, the Talmud Babli and Yerushalmi, and the Midrashic literature*. Judaica Press.

Dror Kamir, Naama Soreq, and Yoni Neeman. 2002. A Comprehensive NLP System for Modern Standard Arabic and Modern Hebrew. In *Proceedings of the ACL-02 Workshop on Computational Approaches to Semitic Languages*, SEMITIC '02, pages 1–9, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yael Dahan Netzer and Michael Elhadad. 1998. Generating Determiners and Quantifiers in Hebrew. In *Proceedings of the Workshop on Computational Approaches to Semitic Languages*, Semitic '98, pages 89–96, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yael Netzer, Meni Adler, David Gabay, and Michael Elhadad. 2007. Can You Tag the Modal? You Should. In *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*, pages 57–64, Prague, Czech Republic. Association for Computational Linguistics.

Helmut Schmid. 1994. Part-of-speech tagging with neural networks. In *Proceedings of the 15th Conference on Computational Linguistics - Volume 1*, COLING '94, pages 172–176, Stroudsburg, PA, USA. Association for Computational Linguistics.

Khalil Sima'an, Alon Itai, Yoad Winter, Alon Altman, and Noa Nativ. 2001. Building a tree-bank of modern hebrew text. *TAL. Traitement automatique des langues*, 42(2):347–380.

Fred Skolnik and Michael Berenbaum, editors. 2007. *Encyclopaedia Judaica vol. 8*. Encyclopaedia Judaica. Macmillan Reference USA, 2 edition. Brovender Chaim and Blau Joshua and Kutscher Eduard Y. and Breuer Yochanan and Eytan Eli sub v. "Hebrew Language".

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich Part-of-speech Tagging with a Cyclic Dependency Network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wido Van Peursen and Constantijn Sikkel. 2014. Hebrew Text Database ETCBC4. type: dataset.

Concept Tagging for Natural Language Understanding: Two Decadelong Algorithm Development

Jacopo Gobbi
University of Trento
Trento, Italy
jacopo.gobbi@studenti.unitn.it

Evgeny A. Stepanov
VUI, Inc.
Trento, Italy
eas@vui.com

Giuseppe Riccardi
University of Trento
Trento, Italy
giuseppe.riccardi@unitn.it

Abstract

English. Concept tagging is a type of structured learning needed for natural language understanding (NLU) systems. In this task, meaning labels from a domain ontology are assigned to word sequences. In this paper, we review the algorithms developed over the last twenty five years. We perform a comparative evaluation of generative, discriminative and deep learning methods on two public datasets. We report on the statistical variability performance measurements. The third contribution is the release of a repository of the algorithms, datasets and recipes for NLU evaluation.

Italiano. *L'annotazione automatica dei concetti è un tipo di apprendimento strutturato necessario per i sistemi di comprensione del linguaggio naturale (NLU). In questo processo le etichette di un'ontologia di dominio sono assegnate a sequenze di parole. In questo articolo esaminiamo gli algoritmi sviluppati negli ultimi venticinque anni. Eseguiamo una valutazione comparativa dei metodi di apprendimento generativo, discriminatorio e approfondito su due set di dati pubblici. Il secondo contributo é un'analisi della variabilitá delle misure di valutazione. Il terzo contributo è il rilascio di un archivio degli algoritmi, dei sets di dati e delle ricette per la valutazione dell'NLU.*

1 Introduction

The NLU component of a conversational system requires an automatic extraction of concept tags, dialogue acts, domain labels and entities. In this paper we describe and review the algorithm development of the concept tagging (a.k.a. slot filling or entity extraction) task. It aims at computing a sequence of concept units, $C = c_1..c_M$, from a sequence of words in natural language, $W = w_1..w_N$. The task can be seen as a structured learning problem where words are the input and concepts are the output labels. In other words, the objective is to map a sentence (utterance) "*I want to go from Boston to Atlanta on Monday*" to the sequence of domain labels "`null null null null null fromloc.city null toloc.city null depart_date.day_name`", that would allow to identify, for instance that *Boston* is a *departure city* . Difficulties may arise from different factors, such as the variable token span of concepts, the long-distance word dependencies, a large and ever changing vocabulary, or subtle semantic implications that might be hard to capture at a surface level or without some prior context knowledge.

Since the early nineties (Pieraccini and Levin, 1992), the task has been designed as a core component of the natural language understanding process in domain-limited conversational systems. Over the years, algorithms have been developed for generative, discriminative and, more recently, for deep learning frameworks. In this paper, we provide a comprehensive review of the algorithms, their parameters and their respective state-of-the-art performances. We discuss the relative advantages and differences amongst algorithms in terms of performances and statistical variability and the optimal parameter settings. Last but not least, we have designed and provided a repository of the data, algorithms, implementations and parameter settings on two public datasets. The GitHub repository[1] is intended as a reference both for practitioners and for algorithm development researchers.

With the conversational AI gaining popularity, the area of NLU is too vast to mention all relevant

[1] www.github.com/fruttasecca/concept-tagging-with-neural-networks

or even recent studies. Moreover the objective of this paper is to benchmark an important subtask of NLU, concept tagging used by advanced conversational systems. We benchmark generative, discriminative and deep learning approaches to NLU, the work is in-line with the works of (Raymond and Riccardi, 2007; Mesnil et al., 2015; Bechet and Raymond, 2018). Unlike previously mentioned comparative performance analysis, in this paper, we benchmark deep learning architectures and compare them to a generative and traditional discriminative algorithms. To the best of our knowledge, this is the first comprehensive comparison of concept tagging algorithms at this scale on public datasets and shared algorithm implementations (and their parameter settings).

2 Algorithms

Among the algorithms considered for benchmarking, we include a representative from the generative class, the weighted finite state transducers (WFSTs), and two discriminative algorithms: Support Vector Machines (SVMs), Conditional Random Fields (CRFs), and a set of base neural networks architectures and their combinations.

Weighted Finite State Transducers[2] cast concept tagging as a translation problem from words to concepts (Raymond and Riccardi, 2007), and usually consist of two components. The first component transduces words to concepts based on a score that can be either induced from data or manually designed; the second component is a stochastic conceptual language model, which re-scores concept sequences. The two components are composed to perform sequence-to-sequence translation and infer the best sequence using Viterbi algorithm.

Support Vector Machines (SVM) are used within Yamcha tool (Kudo and Matsumoto, 2001) that performs sequence labeling using forward and backward moving classifiers. Automatic labels assigned to preceding tokens are used as dynamic features for the current token's label decision.

Conditional Random Fields (CRF)[3] (Lafferty et al., 2001) is a discriminative model based on a dependency graph G and a set of features. Each feature f_k has an associated weight λ_k. Features are generally hand-crafted and their weights are learned from the training data. Additionally, we experiment with word embeddings as additional features for CRFs (CRF+EMB).

Recurrent Neural Networks (RNN). The first neural network architecture[4] we have considered is an Elman RNN (Elman, 1990; Übeyli and Übeyli, 2012). In RNN, a hidden state depends on the current input and the previous hidden state. The output (label), on the other hand, depends on the new hidden state.

Long-Short Term Memory (LSTM) RNNs (Hochreiter and Schmidhuber, 1997) try to tackle the vanishing gradient problem by introducing a more complex mechanisms to address information propagation and deletion, with the cost of a more complex model with more parameters to train due to the system of gates it uses. The memory of the model is represented by the cell state and the hidden state, which also represents the output for the current token. We experimented with a simple LSTM, an LSTM which receives as input the word embedding concatenated with character embeddings obtained through a convolutional layer (Józefowicz et al., 2016) (LSTM-CHAR-REP), and an LSTM with pre-trained embeddings and dynamic embeddings learned from training data (LSTM-2CH). In LSTM-2CH two separate LSTM modules run in parallel and their outputs are concatenated for each word. Similar to the rest of the deep learning models, the output is then fed to a fully connected layer to map every token to the concept tag space.

Gated Recurrent Units (GRU) (Cho et al., 2014) use a reset and an update gate, which are two vectors of weights that decide what information is deleted (or re-scaled) from the current hidden state and how it will contribute to the new hidden state, which is also the output for the current input. Compared to the LSTM model, this allows to train fewer parameters, but introduces a constraint on memory, since it is also used as an output.

Convolutional Neural Networks (CONV) (Majumder et al., 2017; Kim, 2014) consider each sentence as a matrix of shape (# words in sentence, size of embedding) for convolution using kernels of different sizes to pass over the input sequence token-by-token, bigram by bigram and trigram by trigram. The result of convolution is used as a

[2] We use OpenFST (http://www.openfst.org) and OpenGRM (http://www.opengrm.org) libraries.

[3] We use CRFSUITE (Okazaki, 2007) implementation of CRFs in out experiments.

[4] All neural architectures are implemented within the PyTorch framework (https://pytorch.org)

starting hidden memory for a GRU RNN. GRU RNN is used on embedded tokens and starts with the information on the sequence at a global level.

FC-INIT is similar to CONV. The difference is in the pre-elaboration of the hidden state, which is done by fully connected layers elaborating on the whole sequence.

ENCODER architecture (Cho et al., 2014) casts the problem as a sequence-to-sequence translation and consists of two GRU RNNs. Encoder, the first GRU RNN, encodes the input sequence to a fixed vector (the hidden state). Decoder, another GRU RNN, uses the output of the encoder as a starting hidden state. At each step, the decoder receives the label predicted at the previous step as an input, starting with a special token.

ATTENTION architecture is similar to ENCODER with the addition of an attention mechanism (Bahdanau et al., 2014) on the outputs of the encoder. This allows the network to focus on a specific parts of the input sequence. The attention weights are computed with a single fully connected layer that receives as input the embedding of the current word concatenated to the last hidden state.

LSTM-CRF (Yao et al., 2014; Zheng et al., 2015) is an architecture where the LSTM provides class scores for each token, and the Viterbi algorithm decides on the labels of the sequence at a global level using bigrams and transition probabilities that are trained with the rest of the parameters. We also experimented with a variant that considers character level information (LSTM-CRF-CHAR-REP).

3 Corpora

The evaluation of algorithms is performed on two datasets. The Air Travel Information System (**ATIS**) dataset consists of sentences from users querying for information about flights, departure dates, arrivals, etc. The training set consists of 4,978 sentences, while there are 893 sentences that constitute the test set. The average length of a sentence is around 11 tokens, and there are a total of 127 unique tags (with IOB prefixes). Moreover, the large majority of tokens missing an embedding are either numbers or airport/basis/aircraft codes. The training set has a total of 18 types missing an embedding, and the test set has 9.

The second corpus (**MOVIES**)[5] was produced from NL2SparQL (Chen et al., 2014) corpus semi-automatically aligning SPARQL query values to utterance tokens. The dataset follows the split of the original corpus having 3,338 sentences (with 1,728 unique tokens) and 1,084 sentences (with 1,039 tokens) in the training and test sets, respectively. The average length of a sentence is 6.50 and the OOV rate is 0.24. There are 43 concept tags in the dataset. Given the Google embeddings, once we consider every number as a class *number*, we obtain 66 token types without an embedding for the training set and 26 for the test set.

[5] https://github.com/esrel/NL2SparQL4NLU

Model	Parameters	# Params	F_1
WFST	order 4, kneser ney	(7907 states, 842178 arcs)	82.96
	order 4, kneser ney	(4124 states, 76000 arcs)	93.08
SVM	(4, 4) window of tokens, (-1, 0) of POS tag and prefix. Postfix and lemma of current word. Previous two labels.	10364	83.74
	(6, 4) window of tokens, (-1, 0) of prefix and postfix. Previous two labels .	16361	92.91
CRF	(4, 4) window of token, (-1, 0) of POS tag and prefix. Postfix and lemma of current word. Previous + current word conjunction, current + next word conjunction. Bigram model.	1,200K	83.80
	(6, 4) window of tokens, (-1, 0) of prefix. Postfix of current word. Previous + current word conjunction. Bigram model.	2,201K	93.98
CRF+EMB	all above + (4, 4) word embs + current token char embeddings	1,390K	85.85
	all above + (6, 4) word embs + current token char embeddings	3,185K	94.00

Table 1: F_1-scores for the WFST, SVM and CRF (with and without embeddings) algorithms on the MOVIES (top row) and ATIS (bottom row) datasets.

4 Performance Analysis

One of our first observations is the fact that models such as WFST, SVM and CRF yield competitive results with simple setups and few hyperparameters to be tuned. The training of our deep learning models and the search of their hyperparameters would have been unfeasible without dedicated hardware, while it took a fraction of the effort for WFST, SVM and CRF. Moreover, adding word embeddings as features to the CRF allowed it to outperform most of the deep neural networks.

Model	hidden	epochs	batch size	lr	drop rate	emb norm	# of params	min F_1	avg F_1	best F_1
RNN	200	15	50	0.001	0.30	4	1,264K	81.00	82.55	83.96
	400	10	50	0.001	0.25	2	580K	91.80	93.79	95.03
LSTM	200	15	20	0.001	0.70	6	1,505K	82.67	83.76	84.57
	200	15	10	0.001	0.50	8	675K	87.82	94.53	95.36
LSTM-CHAR-REP	400	20	20	0.001	0.70	4	2,085K	82.00	84.28	85.41
	400	15	10	0.001	0.50	6	1,272K	81.00	94.19	95.39
LSTM-2CH	200	20	15	0.001	0.30	8	1,310K	81.22	82.68	83.76
	400	10	100	0.010	0.70	6	1,022K	93.10	94.61	95.38
GRU	200	20	20	0.001	0.50	4	1,424K	76.56	84.29	85.47
	100	15	10	0.005	0.50	10	446K	91.53	94.28	95.28
CONV	200	20	20	0.001	0.50	4	2,646K	84.05	85.02	86.17
	100	15	10	0.005	0.00	2	625K	91.51	94.22	95.38
FC-INIT	100	30	20	0.001	0.30	4	2,805K	82.22	83.93	84.95
	400	15	50	0.010	0.25	4	7,144K	87.39	94.67	95.39
ENCODER	200	30	20	0.001	0.70	4	1,559K	71.25	76.39	79.00
	200	25	5	0.001	0.70	6	730K	70.01	78.16	80.85
ATTENTION	200	15	20	0.001	0.30	4	1,712K	71.86	79.77	82.67
	200	25	5	0.001	0.25	10	894K	92.47	94.09	94.98
LSTM-CRF	200	10	1	0.001	0.70	6	1,507K	84.75	**86.11**	87.47
	400	15	10	0.001	0.50	6	1,200K	94.39	94.72	95.01
LSTM-CRF-CHAR-REP	200	15	1	0.001	0.70	8	1,555K	85.07	86.08	87.05
	200	20	5	0.001	0.50	4	740K	94.45	**94.91**	95.12

Table 2: All models are bidirectional and have been trained with unfrozen Google embeddings, except for CONV and LSTM-2CH. Min, average and best F_1 scores are obtained training the same model with the same hyperparameters, but different parameter initializations. Averages are from 50 runs for MOVIES and 25 for ATIS. For each architecture, the first row reports F_1-score for the MOVIES dataset and the second for ATIS. Hyperparameter search has been done randomly over ranges of values taken from published work. The number of parameters refers to the network parameters plus the embeddings, when those are unfrozen. Given a hidden layer size X reported in **hidden** column, each component in the bidirectional architecture would have a hidden layer size of $X/2$. Similarly, each of the two LSTM components in the LSTM-2CH model would have $X/2$ as a hidden layer size; and each bidirectional component would thus have a hidden layer size equal to $X/4$.

We attribute this to two factors: (1) since these models, unlike neural networks, do not learn feature representation from data, they are simpler and faster to train; and, most importantly, (2) these models usually perform global optimization over the label sequence, while neural networks usually do not. Augmenting neural networks with CRF is not expensive in terms of parameters. Having a CRF component on top of an LSTM increments the number of parameters up to the square of the tag-set size (about 2,500 for the MOVIES dataset), and provides the best performing model.

There seems to be no strong correlation between the number of parameters and the variance of a model performance with respect to the random initialization of its parameters. This is surprising, given the intuition that more parameters can potentially lead to a lower probability of being stuck in a local minima. The case may be that different initializations lead to different training times required to get to good local minimas.

4.1 Statistical Significance Testing

The best performing algorithms in our experimental settings are LSTM-CRF and LSTM-CRF-CHAR-REP; however, they are not very far from CRF+EMB and CRF algorithms. In order to compare the performances in terms of statistical significance, we perform Welch's unequal variances t-test (Welch, 1947), which, compared to more popular Student's t-test, does not assume equal variances. The choice of test is motivated by the observation that neural architectures generally yield higher variances than, for instance, CRF.

The performances are compared on 10-fold cross-validation outputs on the training set for both ATIS and MOVIES datasets. Due to the higher variance of neural network architectures, a better way to test would be to perform many runs with different random initializations for each fold, and take the average of these results; however, such a procedure is computationally very demanding.

ALGORITHMS	*CRF*	*CRF-EMB*	*LSTM-CRF*	*LSTM-CRF-CHAR-REP*
MOVIES				
CRF				
CRF-EMB	*			
LSTM-CRF	*			
LSTM-CRF-CHAR-REP	*			
ATIS				
CRF				
CRF-EMB				
LSTM-CRF	*			
LSTM-CRF-CHAR-REP	*	*		

Table 3: Results of statistical significance testing using Welch's t-test for MOVIES and ATIS datasets. Algorithms on rows with statistically significant differences in performance with $p < 0.05$ in comparison to the algorithms on columns are marked with '*'.

The results of the statistical significance testing are reported in Table 3. For the MOVIES dataset, all the compared models (CRF-EMB, LSTM-CRF, LSTM-CRF-CHAR-REP) significantly outperform the CRF model with $p < 0.05$. However, these models do not yield statistically significant differences among themselves. Specifically, using embeddings with CRF (i.e. CRF-EMB) produces statistically significant differences in performance on top of CRF. Using CRF with LSTM, even though produces better average F_1 than CRF-EMB, the gain is not statistically significant, irrespective of the type of embeddings used.

For the ATIS dataset, on the other hand, use of embeddings with CRF does not yield statistically significant differences with respect to plain CRF. Neural architectures (LSTM-CRF and LSTM-CRF-CHAR-REP), on the other hand, do produce statistically significant difference in performance in comparison to CRF. Moreover, unlike for MOVIES dataset, the use of character embeddings in LSTM-CRF architecture significantly outperforms the CRF-EMB model.

4.2 Error Analysis

Both MOVIES and ATIS datasets have imbalanced distribution of concept labels. The imbalanced distribution of labels is known to affect the performance of the minority classes. Consequently, we correlate the distribution of labels in the training set to the percent of their mis-labeling in the test set (by any model). As expected, the mis-labeling chance is inversely correlated to the percentage of instances the label has in the training set (e.g. given that a label amounts to less than 1% of a dataset, it usually has a mis-labeling chance greater than 10%). For both datasets, the Kendall rank correlation coefficients (Kendall, 1938) are approximately 0.6.

Independent of the distribution, there are certain concepts that are mis-labeled more often. For example, this is the case for **producer name**, **person name**, and **director name** in MOVIES, and **city name**, **state name**, and **airport name** in ATIS. It is not surprising given that these concepts share the values (e.g. the same person may be an actor, director, and producer) and frequently lexical contexts.

Supporting the observations in (Bechet and Raymond, 2018) for ATIS, some errors stem from inconsistent labeling. For instance, in the MOVIES dataset, "*classic cars*" is mapped to "`O O`", but "*are there any documentaries on classic cars*" appears as "`O O O B-movie.genre O B-movie.subject I-movie.subject`".

5 Conclusion

One of the main outcomes of our experiments is that sequence-level optimization is key to achieve the best performance. Moreover, augmenting any neural architecture with a CRF layer on top has a very low cost in terms of parameters and a very good return in terms of performance. Our best performing models (in terms of average F_1) are LSTM-CRF and LSTM-CRF-CHAR-REP. In general we may say that adding a sequence level control to different type of NN architectures leads to very good model performances. Another important observation is the variance of performance of NN models with respect to initialization parameters. Consequently, we strongly believe that this variability should be taken into consideration and reported (with the lowest and highest performances) to improve the reliability and replicability of the published results.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Frederic Bechet and Christian Raymond. 2018. Is ATIS too shallow to go deeper for benchmarking spoken language understanding models? In *Interspeech*.

Yun-Nung Chen, Dilek Hakkani-Tür, and Gokan Tur. 2014. Deriving local relational surface forms from dependency-based entity embeddings for unsupervised spoken language understanding. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 242–247. IEEE.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078.

Jeffrey L. Elman. 1990. Finding structure in time. *COGNITIVE SCIENCE*, 14(2):179–211.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November.

Rafal Józefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *CoRR*, abs/1602.02410.

M. G. Kendall. 1938. A new measure of rank correlation. *Biometrika*, 30(1-2):81–93.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751.

Taku Kudo and Yuji Matsumoto. 2001. Chunking with support vector machines. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*, NAACL '01, pages 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.

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.

Navonil Majumder, Soujanya Poria, Alexander Gelbukh, and Erik Cambria. 2017. Deep learning-based document modeling for personality detection from text. *IEEE Intelligent Systems*, 32(2):74–79, March.

Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al. 2015. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):530–539.

Naoaki Okazaki. 2007. Crfsuite: a fast implementation of conditional random fields (crfs). *URL http://www. chokkan. org/software/crfsuite*.

Roberto Pieraccini and Esther Levin. 1992. Stochastic representation of semantic structure for speech understanding. *Speech Communication*, 11(2):283 – 288. Eurospeech '91.

Christian Raymond and Giuseppe Riccardi. 2007. Generative and discriminative algorithms for spoken language understanding. In *INTERSPEECH*, pages 1605–1608. ISCA.

Elif Derya Übeyli and Mustafa Übeyli. 2012. Case studies for applications of elman recurrent neural networks.

B. L. Welch. 1947. The generalization of 'student's' problem when several different population variances are involved. *Biometrika*, 34(1-2):28–35.

Kaisheng Yao, Baolin Peng, Geoffrey Zweig, Dong Yu, Xiaolong(Shiao-Long) Li, and Feng Gao. 2014. Recurrent conditional random field for language understanding. In *ICASSP 2014*. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), January.

Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip H. S. Torr. 2015. Conditional random fields as recurrent neural networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pages 1529–1537, Washington, DC, USA. IEEE Computer Society.

The language-invariant aspect of compounding: Predicting compound meanings across languages

Fritz Günther
University of Milano-Bicocca
Milan, Italy
fritz.guenther@unimib.it

Marco Marelli
University of Milano-Bicocca
Milan, Italy
marco.marelli@unimib.it

Abstract

English. In the present study, we investigated to what extent compounding involves general-level cognitive abilities related to conceptual combination. If that was the case, the compounding mechanism should be largely invariant across different languages. Under this assumption, a compositional model trained on word representations in one language should be able to predict compound meanings in other languages. We investigated this hypothesis by training a word embedding-based compositional model on a set of English compounds, and subsequently applied this model to German and Italian test compounds. The model partially predicted compound meanings in German, but not in Italian.

Italiano. *In questo lavoro abbiamo investigato quanto la composizione sottenda abilità cognitive generali relata alla combinazione concettuale. Se questo fosse il caso, il meccanismo composizionale dovrebbe variare in maniera limitata tra diverse lingue. Di conseguenza, un modello composizionale basato su rappresentazioni lessicali in una data lingua dovrebbe essere in grado di predire significati composizionali in altre lingue. Abbiamo testato questa ipotesi addestrando un modello composizionale sui word embeddings di un set di composti inglesi, e successivamente testato lo stesso modello su composti tedeschi e italiani. Il modello è in grado di predire in modo parzialmente corretto le rappresentazioni dei composti in tedesco, ma non italiano.*

1 Introduction

Compounds are complex words such as *airport*, with two constituents that can be used as free words. Compounding is a highly prevalent phenomenon across many languages. It has been argued to be a proto-linguistic structure to combine simple words into novel and complex concepts, from which more complex compositional language structures have been derived (Jackendoff, 2002).

Given the prevalence and ubiquity of compounding across languages, it is reasonable to assume that speakers of different languages rely, to some degree, on the same cognitive mechanisms to compose the meanings of constituents into a compound meaning. Indeed, the linguistic phenomenon of compounding is generally considered to be the linguistic mirror of the cognitive process of *conceptual combination* (Gagné and Spalding, 2009; Murphy, 2002). Thus, while specific aspects of compounding will inevitably vary between languages due to differences in the language structure and other idiosyncracies, we assume that there is also a language-invariant aspect of compounding that can be transferred across languages. We will investigate this hypothesis by examining whether a compositional model trained on one language (English) is able to predict compound meanings in other languages (German and Italian).

2 Compositional Model

In our study, word meanings are represented via word embeddings derived from large corpora using the *word2vec* model (Mikolov et al., 2013). As a model to derive compound meaning representations from these vectors, we employ the CAOSS model (Marelli et al., 2017), which relies on the compositional model for distributional word vectors proposed by Guevara (2010).

The CAOSS model computes the meaning of a

compound as

$$c = M \cdot u + H \cdot v \quad (1)$$

, where c is the n-dimensional vector representing the compound meaning, u and v are the n-dimensional vectors representing the first and second constituent, respectively, and M and H are $n \times n$-dimensional weight matrices updating the free word meanings into constituent meanings before they are combined.

The weight matrices M and H are estimated through a training procedure on all compound words available in the source corpus for the word embeddings. They are estimated in a least-square regression procedure aimed at optimally predicting these observed compound meanings c from the constituent meanings u and v, following Equation 1.

3 Evaluation Material

In order to investigate our hypothesis, we employed three sets of compounds, collected from various sources: The English set consisted of 5,618 compounds in closed form, collected from the words tagged as noun-noun combinations in the CELEX database (Baayen et al., 1995) and the English Lexicon Project (Balota et al., 2007), and in hyphenated form, collected from the *ukWaC* corpus as described below. The German set consisted of 3,451 compounds in closed form, collected from (Brysbaert et al., 2011) and the GhostNN database (Schulte im Walde et al., 2016). The Italian set of 216 compounds in closed form, collected by one of the authors from an Italian dictionary (Sabatini and Coletti, 2007). Note that the Italian dataset is smaller than the other sets, since compounds are far less common in Italian than in English or German, where compounds are extremely prevalent and compounding is highly productive.
No restrictions based on linguistic criteria (such as endocentric vs. exocentric, or head-first vs. head-second) were applied in the selection of the compounds.

4 Inducing Word Vectors and Training the Compositional Model

4.1 Word Embeddings

Word embeddings were trained on three different web-based corpora (`http://wacky.sslmit.unibo.it`):
The English 2 billion word corpus *ukWaC*, the German 1.7 billion word corpus *deWaC*, and the Italian 2 billion word vorpus *itWaC*. While these corpora are not parallel corpora, they were collected using the same web crawler run on different domains (.uk, .de, and .it, respectively). Furthermore, they are very large corpora, which should lead to highly averaged word meaning representations within all three languages. From each of these corpora, *word2vec* word embeddings were derived using the parameter set shown to produce the best results by Baroni et al. (2014): The *cbow* algorithm with a context window size of 5 words producing 400-dimensional vectors (negative sampling with $k = 10$, subsampling with $t = 1e^{-5}$). Word embeddings were only trained for words that occurred more than 50 times in a source corpus.

4.2 Second-Level Vectors

Obviously, the three different semantic spaces were not comparable to one another, as each set of word vectors was trained only on a single-language corpus. Since the weights specified in the matrices M and H of the CAOSS model encode how much each output dimension value for the constituent-updated vectors Mu and Hv is influenced by each input dimension value of the word vectors for the constituents u and v, we could not reasonably apply the CAOSS model trained in one language to word embeddings in another language. We needed word vectors whose dimensions are comparable across the three languages. To this end, we decided to construct *second-level vectors* from the original word embeddings.

The basis for these second-level vectors is the observation that, while word embeddings are not comparable between languages, the similarity *structure* between sets of words is highly comparable across languages. We exploit this observation to define second-level vectors as vectors of similarities between the target and an ordered list of content words (see Table 1). By choosing a list of content words that are as unambiguous as possible and have clear translations across all three languages (such as *pizza*, *Pizza*, *pizza*), we aimed at keeping the second-level vector entries as comparable as possible across languages. We constructed a list containing 300 such aligned content words. With these words, we can demonstrate

original word embeddings				
	dim1	dim2	dim3	...
tomato$_{en}$	0.58	-0.66	-0.92	....
Tomate$_{de}$	-0.23	0.12	0.20	....
pomodoro$_{it}$	-0.01	0.39	-1.37	....
second-level vectors				
en	red	pizza	horse	...
de	rot	Pizza	Pferd	...
it	rosso	pizza	cavallo	...
tomato$_{en}$	0.22	0.28	0.07	....
Tomate$_{de}$	0.23	0.30	0.12	....
pomodoro$_{it}$	0.23	0.26	0.04	....

Table 1: An example for dimensional values of original and second-level word embeddings.

that the similarity structure between words is indeed comparable across languages: We computed all pairwise similarities between these 300 words within each language, and then compared this list of similarities across languages. Similarity correlations across the three languages are substantial: $r = .77$ for English-German, $r = .76$ for English-Italian, and $r = .79$ for German-Italian.

With this aligned list, we converted our word embeddings into second-level vectors by computing, within each language, the cosine similarities between each word in the original semantic space and the 300 content words (see Table 1).

4.3 Evaluation of Second-Level Vectors

In order to serve as adequate word vectors for our compositional model, these second-level vectors need to satisfy two criteria: Firstly, they must adequately capture the similarity structure of the original word embeddings within each language, in order to be used as a substitute for the original word embeddings. Secondly, they have to align word vectors between the three languages: for example, the second-level vector for *tomato* in English should be very similar to the second-level vector for *Tomate* in German and for *pomodoro* in Italian.

Within-Language Reliability. To test for within-language constancy, we first computed the pairwise cosine similarities between all compound constituents from these item sets. Additionally, we computed the cosine similiarities between each compound and its two constituents within each language. These are valid test sets for our study since these are the very embeddings employed to run and test our compositional model later on. In a next step, we computed the same similarities using not the original word embeddings, but the second-level vectors. We then calculated correlations between all the similarity scores computed from the two different vector sets for each of the three languages.

For English, the correlation between the pairwise constituent similarities (2,386 different constituents) was $r = .86$, and the correlation between the constituent-compound similarities was $r = .79$. For German, the correlation between the pairwise constituent similarities (1,929 different constituents) was $r = .80$, and the correlation between the constituent-compound similarities was $r = .72$. For Italian, the correlation between the pairwise constituent similarities (568 different constituents) was $r = .81$, and the correlation between the constituent-compound similarities was $r = .74$. Thus, the similarity structure of the original semantic spaces is to a large extent captured by the second-level vectors, which qualifies them as reliable word meaning representations for our study.

Between-Language Alignment. We tested the across-language alignment of the second-level vectors by means of the original list of 300 content words. This list was constructed to include words that have single clear translation across all three languages. Thus, if the second-level vectors are indeed aligned across the three languages, the three vectors representing these translated words in each language should be very similar to one another.

To test this, we computed the cosine similarity between each of the three translations of these words across the three languages. Using the original word embeddings, the average similarities were virtually zero, as expected for different model trained on different languages: $M = .01$ for English-German, $M = -.00$ for English-Italian, and $M = .01$ for German-Italian. However, computing the same similarities from the second-level vectors improved results dramatically: $M = .80$ for English-German, $M = .80$ for English-Italian, and $M = .82$ for German-Italian. Thus, the second-level vectors are to a large extent aligned across languages, providing the ground to apply a composition model trained on vectors in one language on vectors of the other languages.

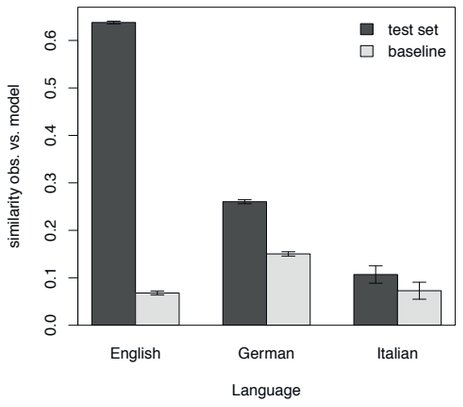


Figure 1: Similarities (mean values and .95 confidence intervals) between observed and model-derived (second-level) vectors for compounds across the three different languages.

4.4 Training the CAOSS model

The CAOSS model was trained on the English second-level word vectors. As a training set, we employed the set of 5,618 English compounds described in the section *Evalutation Material*. The other two languages, German and Italian, were not considered during training.

5 Results

Using the matrices M and H obtained from this training, we computed, for each compound in our evaluation sets, its compound meaning as predicted from the compositional CAOSS model (see Equation 1). The model trained on English was used to compute the model-derived compound meanings for all three languages. We then computed the cosine similarities between these predicted meanings and the corresponding, actually "observed" compound meanings (their respective second-level vectors; e.g. *airport* – [*air+port*]). As a baseline comparison level within each language, we computed similarities between the observed compound meanings and model-derived meanings for a random pair of nouns (such as *airport* – [*spring+feeling*]). The mean similarities are displayed in Figure 1.

For English, on which our CAOSS model was trained, we obtained a mean similarity between model-derived and observed vectors of $M = .64$, which was significantly above the random baseline ($t(5617) = 122.4, p < .001$).

For the German evaluation set, the mean similarity between model-derived and observed vectors was $M = .26$, which is significantly above baseline ($t(3450) = 20.12, p < .001$).

In contrast, for the Italian evaluation set, the actual similarities did not beat the baseline ($t(215) = 1.39$, $p = .165$). Note that Italian compounds can be classified into head-first compounds (such as *pescespada* – *swordfish*, lit. *fishsword*) or head-second compounds (such as *funivia* – (lit.) *ropeway*)[1]. However, the actual similarities did not beat the baseline in either case ($t(58) = 1.67$, $p = .100$ for head-first compounds; $t(156) = 0.56, p = .578$ for head-second compounds).

The mean value in English differed significantly from German ($t(6460) = 75.53$, $p < .001$), which in turn differed significantly from Italian ($t(238) = 8.18, p < .001$).

6 Discussion

Our results show that a compositional model trained in one language exclusively (English) can be applied to another language (German) to partially predict the meanings of compounds in the latter, of which the model had no training experience at all. Obviously, the model trained on English compounds predicted English compound meanings far better than German compound meanings. This does not stand contrary to our hypothesis: We do not assume that compounding is a tout-court language-invariant mechanism, but that compounding also encompasses general mechanisms *besides* language-specific features.

However, the model trained on English was not able to predict Italian compound meanings above baseline level. Thus, our results only partially support our hypothesis. In interpreting this finding, it has to be considered that the Italian evaluation set was far smaller than the English and the German sets, leading to decreased statistical power in this case (note that, on a purely descriptive level, model performance in Italian is slightly above baseline). Keeping that in mind, our results indicate that the applicability of a compositional model across languages seems to depend on the similarity between the language in which a model was trained and the one where it is applied.

[1]The head is the compound constituent that denotes the semantic category of a word: an *airport* is a type of *port*.

In structural terms, German is in fact much more similar to English than Italian. Both English and German are West-Germanic languages which almost exclusively produce head-second compounds and have highly productive and very rich compounding systems. Italian compounds however can be head-first or head-second, and the compounding system is far less productive in Italian than in English or German (one of the factors responsible for the fact that our Italian item set was smaller than the English or German sets). This explanation is still tentative given the restricted range of languages investigated here. A more thorough investigation on this specific issue would require tests on a wide range of languages, which should be theoretically characterized in terms of their structural similarity with respect to compounding beforehand.

Additionally, future work is required to address other language-dependent aspects of compounding. For example, we focussed only on closed-form compounds, while some languages (for example English and Italian, but not German) can produce open forms such as *school bus* or *pesce spada*. Another issue to be investigated more closely is headedness. On the one hand, head-second Italian compounds are more similar to English and German from a structural point of view; on the other hand, head-first compounds are assumed to be more like English and German in terms of productivity and regularity of meaning. Although our item set included head-first as well as head-second Italian compounds, both are obviously still smaller than the complete Italian item set. Thus, in future studies larger item sets are required to provide such differential tests with the necessary statistical power.

Acknowledgments

This work was supported by Research Fellowship 392225719 from the German Research Foundation (DFG), awarded to Fritz Günther.

References

R. H. Baayen, R. Piepenbrock, and L. Gulikers. 1995. The CELEX lexical data base (CD-ROM).

David A Balota, Melvin J Yap, Keith A Hutchison, Michael J Cortese, Brett Kessler, Bjorn Loftis, James H Neely, Douglas L Nelson, Greg B Simpson, and Rebecca Treiman. 2007. The English Lexicon Project. *Behavior Research Methods*, 39:445–459.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of ACL 2014*, pages 238–247, East Stroudsburg, PA. ACL.

Marc Brysbaert, Matthias Buchmeier, Markus Conrad, Arthur M Jacobs, Jens Bölte, and Andrea Böhl. 2011. The word frequency effect. *Experimental psychology*, 58:412–424.

Christina L. Gagné and Thomas L. Spalding. 2009. Constituent integration during the processing of compound words: Does it involve the use of relational structures? *Journal of Memory and Language*, 60:20–35.

Emiliano Guevara. 2010. A regression model of adjective-noun compositionality in distributional semantics. In *Proceedings of the 2010 Workshop on Geometrical Models of Natural Language Semantics*, pages 33–37. Association for Computational Linguistics.

Ray Jackendoff. 2002. *Foundations of Language: Brain, Meaning, Grammar, Evolution*. Oxford University Press, Oxford, UK.

Marco Marelli, Christina L. Gagné, and Thomas L. Spalding. 2017. Compounding as abstrat operation in semantic space: A data-driven, large-scale model for relational effects in the processing of novel compounds. *Cognition*, 166:207–224.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv:1301.3781v3*.

Gregory L. Murphy, 2002. *Conceptual Combination*, pages 443–475. MIT Press, Cambridge, MA.

Francesco Sabatini and Vittorio Coletti. 2007. *Dizionario della lingua italiana*. RCS Libri, Milano, Italy.

Sabine Schulte im Walde, Anna Hätty, Stefan Bott, and Nana Khvtisavrishvili. 2016. G_hoSt-NN: A Representative Gold Standard of German Noun-Noun Compounds. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 2285–2292, Portoroz, Slovenia.

From General to Specific: Leveraging Named Entity Recognition for Slot Filling in Conversational Language Understanding

Samuel Louvan
University of Trento
Fondazione Bruno Kessler
slouvan@fbk.eu

Bernardo Magnini
Fondazione Bruno Kessler
magnini@fbk.eu

Abstract

English. Slot filling techniques are often adopted in language understanding components for task-oriented dialogue systems. In recent approaches, neural models for slot filling are trained on domain-specific datasets, making it difficult porting to similar domains when few or no training data are available. In this paper we use multi-task learning to leverage general knowledge of a task, namely Named Entity Recognition (NER), to improve slot filling performance on a semantically similar domain-specific task. Our experiments show that, for some datasets, transfer learning from NER can achieve competitive performance compared with the state-of-the-art and can also help slot filling in low resource scenarios.

Italiano. *Molti sistemi di dialogo task-oriented utilizzano tecniche di slot-filling per la comprensione degli enunciati. Gli approcci piú recenti si basano su modelli neurali addestrati su dataset specializzati per un certo dominio, rendendo difficile la portabilitá su dominii simili, quando pochi o nessun dato di addestramento é disponibile. In questo contributo usiamo multi-task learning per sfruttare la conoscenza generale proveniente da un task, precisamente Named Entity Recognition (NER), per migliorare le prestazioni di slot filling su dominii specifici e semanticamente simili. I nostri esperimenti mostrano che transfer learning da NER aiuta lo slot filling in dominii con poche risorse e raggiunge risultati competitivi con lo stato dell'arte.*

1 Introduction

In dialogue systems, semantic information of an utterance is generally represented with a *semantic frame*, a data structure consisting of a domain, an intent, and a number of slots (Tur, 2011). For example, given the utterance *"I'd like a United Airlines flight on Wednesday from San Francisco to Boston"*, the domain would be **flight**, the intent is **booking**, and the slot fillers are **United Airlines** (for the slot `airline_name`), **Wednesday** (`booking_time`), **San Francisco** (`origin`), and **Boston** (`destination`). Automatically extracting this information involves domain identification, intent classification, and slot filling, which is the focus of our work.

Slots are usually domain specific as they are predefined for each domain. For instance, in the flight domain the slots might be `airline_name`, `booking_time`, and `airport_name`, while in the bus domain the slots might be `pickup_time`, `bus_name`, and `travel_duration`. Recent successful approaches related to slot filling tasks (Wang et al., 2018; Liu and Lane, 2017a; Goo et al., 2018) are based on variants of recurrent neural network architecture. In general there are two ways of approaching the task: (i) by training a single model for each domain; or (ii) by performing domain adaptation, which results in a model that learns better feature representations across domains. All these approaches directly train the models on domain-specific slot filling datasets.

In our work, instead of using a domain-specific slot filling dataset, which can be expensive to obtain being task specific, we propose to leverage knowledge gained from a more "general", but semantically related, task, referred as the *auxiliary task*, and then transfer the learned knowledge to the more specific task, namely slot filling, referred as the *target task*, through transfer learning. In the literature, the term transfer learning can be used

in different ways. We follow the definition from (Mou et al., 2016), in which transfer learning is viewed as a paradigm which enables a model to use knowledge from auxiliary tasks to help the target task. There are several ways to train this model: we can directly use the trained parameters of the auxiliary tasks to initialize the parameters in the target task (*pre-train* & *fine-tuning*), or train a model of auxiliary and target tasks simultaneously, where some parameters are shared (*multi-task learning*).

We propose to train a slot filling model jointly with Named Entity Recognition (NER) as an auxiliary task through multi-task learning (Caruana, 1997). Recent studies have shown the potential of multi-task learning in NLP models. For example, (Mou et al., 2016) empirically evaluates transfer learning in sentence and question classification tasks. (Yang et al., 2017) proposes an approach for transfer learning in sequence tagging tasks.

NER is chosen as the auxiliary task for several reasons. First, named entities frequently occur as slot values in several domains, which make them a relevant general knowledge to exploit. The same NER type can refer to different slots in the same utterance. On the previous utterance example, the NER labels are `LOC` for both *San Francisco* and *Boston*, and `ORG` for *United Airlines*. Second, state-of-the-art performance of NER (Lample et al., 2016; Ma and Hovy, 2016) is relatively high, therefore we expect that the transferred feature representation can be useful for slot filling tasks. Third, large annotated NER corpora are easier to obtain compared to domain-specific slot filling datasets.

The contributions of this work are as follows: we investigate the effectiveness of leveraging Named Entity Recognition as an auxiliary task to learn general knowledge, and transfer this knowledge to slot filling as the target task in a multi-task learning setting. To our knowledge, there is no reported work that uses NER transfer learning for slot filling in conversational language understanding. Our experiments show that for some datasets multi-task learning achieves better overall performance compared to previous published results, and performs better in some low resource scenarios.

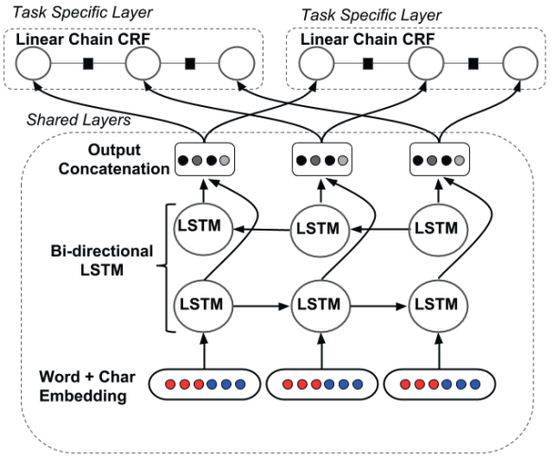


Figure 1: Multi-task Learning Network architecture.

2 Related Work

Recent approaches on slot filling for conversational agents are based mostly on neural models. The work by (Wang et al., 2018) introduces a bi-model Recurrent Neural Network (RNN) structure to consider cross-impact between intent detection and slot filling. (Liu and Lane, 2016) propose an attention mechanism on the encoder-decoder model for joint intent classification and slot filling. (Goo et al., 2018) extends the attention mechanism using a slot gated model to learn relationships between slot and intent attention vectors. The work from (Hakkani-Tür et al., 2016) uses bidirectional RNN as a single model that handles multiple domains by adding a final state that contains domain identifier. (Jha et al., 2018; Kim et al., 2017) uses expert based domain adaptation while (Jaech et al., 2016) proposes a multi-task learning approach to guide the training of a model for new domains. All of these studies train their model solely on slot filling datasets, while our focus is to leverage more "general" resources, such as NER, by training the model simultaneously with slot filling through multi-task learning.

3 Model

In this Section we describe the base model that we use for the slot filling task and the transfer learning model between NER and slot filling.

3.1 Base Model

The model that we use is a hierarchical neural based model, as it has shown to be the state of the art in sequence tagging tasks such as named entity recognition (Ma and Hovy, 2016; Lample

Sentence	find	flights	from	Atlanta	to	Boston
Slot	O	O	O	B-fromloc	O	B-toloc

Table 1: An example output from the model.

et al., 2016). Figure 1 depicts the overall architecture of the model. The model consists of several stacked bidirectional RNNs and a CRF layer on top to compute the final output. The input of the model are both words and characters in the sentence. Each word is represented with a word embedding, which is simply a lookup table. Each word embedding is concatenated with its character representation. The character representation itself can be composed from a concatenation of the final state of bidirectional LSTM (Hochreiter and Schmidhuber, 1997) over characters in a word or extracted using a Convolutional Neural Network (CNN) (LeCun et al., 1998). The concatenation of word and character embeddings is then passed to a LSTM cell. The output of the LSTM in each time step is then fed to a CRF layer. Finally, the output of the CRF layer is the slot tag for a word in the sentence, as shown in Table 1.

3.2 Transfer Learning Model

In the context of NLP, recent studies have applied transfer learning in tasks such as POS tagging, NER, and semantic sequence tagging (Yang et al., 2017; Alonso and Plank, 2017). In general, a popular mechanism is to do multitask learning with a network that optimizes the feature representation for two or more tasks simultaneously. In particular, among the tasks we can set target tasks and auxiliary tasks. In our case, the target task is the slot filling task and the auxiliary task is the NER task. Both tasks are using the base model explained in the previous section with a task specific CRF layer on top.

4 Experimental Setup

The objective of our experiment is to validate the hypothesis that by training a slot filling model with semantically related tasks, such as NER, can be helpful to the slot filling performance. We compare the performance of Single Task Learning (STL) and Multi-Task Learning (MTL). STL uses the Bi-LSTM + CRF model described in (§3.1) and it is trained directly on the target slot filling task. MTL refers to (§3.2), in which models for slot filling and NER are trained simultaenously and some parameters are shared.

Dataset	#sents	#tokens	#label	Label Examples
Slot Filling				
ATIS	4478	869	79	airport name, airline name, return date
MIT Restaurant	6128	3385	20	restaurant name, dish, price, hours
MIT Movie	7820	5953	8	actor, director, genre, title, character
NER				
CoNLL 2003	14987	23624	4	person, location, organization
OntoNotes 5.0	34970	39490	18	organization, gpe, date, money, quantity

Table 2: Training data statistics.

Data. We use three conversational slot filling datasets to evaluate the performance of our approach: the ATIS dataset on Airline Travel Information Systems (Tür et al., 2010), the MIT Restaurant and the MIT Movie datasets[1] (Liu et al., 2013; Liu and Lane, 2017a) on restaurant reservations and movie information respectively. Each dataset provides a number of conversational user utterances, where tokens in the utterance are annotated with their domain specific slot. As for the NER dataset, we use two datasets: CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003) and Ontonotes 5.0 (Pradhan et al., 2013). For OntoNotes, we use the Newswire section for our experiments. Table 2 shows the statistics and example labels of each dataset. We use the training-test split provided by the developers of the datasets, and have further split the training data into 80% training and 20% development sets.

Implementation. We use the multi-task learning implementation from (Reimers and Gurevych, 2017) and have adapted it for our experiments. We consider slot filling as the target task and NER as the auxiliary task. We use a pretrained embedding

[1] https://groups.csail.mit.edu/sls/downloads/

Model	ATIS	MIT Restaurant	MIT Movie
Bi-model based (Wang et al., 2018)	**96.89**	-	-
Slot gated model (Goo et al., 2018)	95.20	-	-
Recurrent Attention (Liu and Lane, 2016)	95.78	-	-
Adversarial (Liu and Lane, 2017b)	95.63	74.47	85.33
Base model (STL)	95.68	78.58	**87.34**
MTL with CoNLL 2003	95.43	78.82	87.31
MTL with OntoNotes	95.78	**79.81**††	87.20
MTL with CoNLL 2003 + OntoNotes	95.69	78.52	86.93

Table 3: F1 score comparison of MTL, STL and the state of the art approaches. †† indicates significant improvement over STL baseline with $p < 0.05$ using approximate randomization testing.

Slot	ATIS		MIT Restaurant		MIT Movie	
	STL	MTL	STL	MTL	STL	MTL
PER	-	-	-	-	**90.73**	89.58
LOC	98.91	**99.32**	81.95	**83.47**††	-	-
ORG	100.00	100.00	-	-	-	-

Table 4: Performance on slots related to CoNLL tags on the development set (MTL with CONLL).

Dataset	#training sents	STL	MTL-C	MTL-O
ATIS	200	84.37	83.15	**84.97**
	400	**87.04**	86.54	86.93
	800	90.67	91.15	**91.58**††
MIT Restaurant	200	54.65	**56.95**††	56.79
	400	62.91	**63.91**	62.29
	800	68.15	**68.52**	68.47
MIT Movie	200	69.97	**71.11**††	69.78
	400	**75.88**	75.23	75.18
	800	79.33	**80.28**††	78.65

Table 5: Performance comparison on low resource scenarios. MTL-C and MTL-O are MTL models trained on CoNLL and OntoNotes datasets respectively. †† indicates significant improvement over STL with $p < 0.05$ using approximate randomization testing.

from (Komninos and Manandhar, 2016) to initialize the word embedding layer. We did not tune the hyperparameters extensively, although we followed the suggestions in a comprehensive study of hyperparameters in sequence labeling tasks from (Reimers and Gurevych, 2017). The word and character embedding dimensions, and dropout rate are set to 300, 30, and 0.25 respectively. The LSTM size is set to 100 following (Lample et al., 2016). We use CNN to generate the character embedding as in (Ma and Hovy, 2016). For each epoch in the training, we train both the target task and the auxiliary task and keep the data size between them proportional. We train the network using Adam (Kingma and Ba, 2014) optimizer. Each model is trained for 50 epochs with early stopping on the target task. We evaluate the performance of the target task by computing the F1-score of the test data following the standard CoNLL-2000 evaluation[2].

5 Results and Analysis

Overall performance. Table 3 shows the comparison of our Single Task Learning (STL) and Multi-Task Learning (MTL) models with the current state of the art performance for each dataset. For the ATIS dataset, the performance of the STL model is comparable to most of the state-of-the-art approaches, however not all MTL models lead to an increase in the performance. As for the MIT Restaurant, both STL and MTL models achieve better performance compared to the previously published results (Liu and Lane, 2017a). For the MIT movie dataset, STL achieves better results by a small margin over MTL. Both STL and MTL performs better than the previous approach for the MIT movie dataset. When we combine CoNLL and OntoNotes into three tasks in the MTL setting, the overall performance tends to decrease across datasets compared to MTL with OntoNotes only.

Per slot performance. Although the overall performance using MTL is not necessarily helpful, we analyze the per slot performance in the development set to get better understanding of the model's behaviour. In particular, we want to know whether slots that are related to CoNLL tags perform better through MTL compared to STL, as evidence of transferable knowledge. To this goal, we manually created a mapping between NER CoNLL tags and slot tags for each dataset. For example in the ATIS dataset, some of the slots that are related to the LOC tags are `fromloc.airport_name` and `fromloc.city_name`. We compute the micro-F1 scores for the slots based on this mapping. Table 4 shows the performance of the slots related to CoNLL tags on the development set. For the ATIS and MIT Restaurant datasets we can see that MTL improves the performance in recognizing `LOC` related tags. While for the MIT Movie dataset, MTL suffers from performance decrease on `PER` tag. There are three slots related to `PER` in MIT Movie namely `CHARACTER`, `ACTOR`, and `DIRECTOR`. We found that the decrease is on `DIRECTOR` while for `ACTOR` and `CHARACTER` there is actually an improvement. We sample 10 sentences in which the model makes mistakes on `DIRECTOR` tag. Of these sentences, four sentences are wrongly annotated. Another four sentences are errors by the model although the sentence seems easy, typically the model is confused between `DIRECTOR` and `ACTOR`. The rests are difficult sentences. For example, the sentence: "*Can you name Akira Kurusawas first color film*". This sentence is somewhat general and the model needs more information to discriminate between `ACTOR` and `DIRECTOR`.

[2]https://www.clips.uantwerpen.be/conll2000/chunking/output.html

Low resource scenario. In Table 5 we compare STL and MTL under varying numbers of training sentences to simulate low resource scenarios. We did not perform MTL including *both* CoNLL and OntoNotes, as the results from Table 3 show that performance tends to degrade when we include both resources. For the MIT Restaurant, for all the low resource scenarios, MTL consistently gives better results. In the MIT Restaurant dataset, it is evident that the less number of training sentences that we have, the more helpful is MTL. For the ATIS and MIT Movie, MTL performs better than STL except for the 400 sentence training scenario. We suspect that to have a more consistent MTL improvement in different low resource scenarios, a different training strategy is needed. In our current experiments, the number of training data is proportional between the target task and auxiliary task. In the future, we would like to try other training strategies, such as using the full training data from the auxiliary task. As the data from the target task is much smaller, we plan to repeat the batch of the target task until we finish training all the batches from the auxiliary task in an epoch. This strategy is similar to (Jaech et al., 2016).

Regarding the variation of results that we get from CoNLL or OntoNotes, we believe that selecting promising auxiliary tasks, or selecting data from a particular auxiliary task, are important to alleviate *negative transfer*. This also has been shown empirically in (Ruder and Plank, 2017; Bingel and Søgaard, 2017). Another alternative to reduce negative transfer, which would be interesting to try in the future, is by using a model which can decide which knowledge to share (or not to share) among tasks (Ruder et al., 2017; Meyerson and Miikkulainen, 2017).

6 Conclusion

In this work we train a slot filling domain-specific model adding NER information, under the assumption that NER introduces useful "general" labels, and that it is cheaper to obtain compared to task specific slot filling datasets. We use multi-task learning to leverage the learned knowledge from NER to slot filling task. Our experiments show evidence that we can achieve comparable or better performance against the state-of-the-art approaches and against single task learning, both in full training data and low resource scenarios. In the future, we are interested in working on datasets in Italian and explore more sophisticated multi-task learning strategies.

Acknowledgments

We would like to thanks three anonymous reviewers and Simone Magnolini, Marco Guerini, Serra Sinem Tekiroğlu for helpful comments and feedback. This work was supported by the grant of Fondazione Bruno Kessler PhD scholarship.

References

Héctor Martínez Alonso and Barbara Plank. 2017. When is multitask learning effective? semantic sequence prediction under varying data conditions. In *15th Conference of the European Chapter of the Association for Computational Linguistics*.

Joachim Bingel and Anders Søgaard. 2017. Identifying beneficial task relations for multi-task learning in deep neural networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 164–169. Association for Computational Linguistics.

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28:41–75.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 753–757.

Dilek Z. Hakkani-Tür, Gökhan Tür, Asli Çelikyilmaz, Yun-Nung Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. 2016. Multi-domain joint semantic frame parsing using bi-directional rnn-lstm. In *INTERSPEECH*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Aaron Jaech, Larry P. Heck, and Mari Ostendorf. 2016. Domain adaptation of recurrent neural networks for natural language understanding. In *INTERSPEECH*.

Rahul Jha, Alex Marin, Suvamsh Shivaprasad, and Imed Zitouni. 2018. Bag of experts architectures for model reuse in conversational language understanding. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, volume 3, pages 153–161.

Young-Bum Kim, Karl Stratos, and Dongchan Kim. 2017. Domain attention with an ensemble of experts. In *ACL*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Alexandros Komninos and Suresh Manandhar. 2016. Dependency based embeddings for sentence classification tasks. In *HLT-NAACL*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of NAACL-HLT*, pages 260–270.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Interspeech 2016*.

Bing Liu and Ian Lane. 2017a. Multi-domain adversarial learning for slot filling in spoken language understanding. In *NIPS Workshop on Conversational AI*.

Bing Liu and Ian Lane. 2017b. Multi-Domain Adversarial Learning for Slot Filling in Spoken Language Understanding.

Jingjing Liu, Panupong Pasupat, Yining Wang, Scott Cyphers, and James R. Glass. 2013. Query understanding enhanced by hierarchical parsing structures. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, December 8-12, 2013*, pages 72–77.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1064–1074.

Elliot Meyerson and Risto Miikkulainen. 2017. Beyond shared hierarchies: Deep multitask learning through soft layer ordering. *arXiv preprint arXiv:1711.00108*.

Lili Mou, Zhao Meng, Rui Yan, Ge Li, Yan Xu, Lu Zhang, and Zhi Jin. 2016. How transferable are neural networks in nlp applications? *arXiv preprint arXiv:1603.06111*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using ontonotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152.

Nils Reimers and Iryna Gurevych. 2017. Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 338–348, Copenhagen, Denmark, 09.

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 Ruder, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. 2017. Learning what to share between loosely related tasks. *arXiv preprint arXiv:1705.08142*.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Gökhan Tür, Dilek Z. Hakkani-Tür, and Larry P. Heck. 2010. What is left to be understood in atis? *2010 IEEE Spoken Language Technology Workshop*, pages 19–24.

Gokhan Tur. 2011. *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*. John Wiley and Sons, New York, NY, January.

Yu Wang, Yilin Shen, and Hongxia Jin. 2018. A bi model based rnn semantic frame parsing model for intent detection and slot filling. In *NAACL*.

Zhilin Yang, Ruslan Salakhutdinov, and William W Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. *arXiv preprint arXiv:1703.06345*.

What's in a Food Name: Knowledge Induction from Gazetteers of Food Main Ingredient

Bernardo Magnini[1], **Vevake Balaraman**[1,2], **Simone Magnolini**[1,3], **Marco Guerini**[1]
[1] Fondazione Bruno Kessler, Via Sommarive 18, Povo, Trento — Italy
[2] University of Trento, Italy. [3] AdeptMind Scholar
{magnini, balaraman, magnolini, guerini}@fbk.eu

Abstract

English. We investigate head-noun identification in complex noun-compounds (e.g. *table* is the head-noun in *three legs table with white marble top*). The task is of high relevancy in several application scenarios, including utterance interpretation for dialogue systems, particularly in the context of e-commerce applications, where dozens of thousand of product descriptions for several domains and different languages have to be analyzed. We define guidelines for data annotation and propose a supervised neural model that is able to achieve 0.79 F1 on Italian food noun-compounds, which we consider an excellent result given both the minimal supervision required and the high linguistic complexity of the domain.

Italiano. *Affrontiamo il problema di identificare head-noun in nomi composti complessi (ad esempio "tavolo" is the head-noun in "tavolo con tre gambe e piano in marmo bianco"). Il compito é di alta rilevanza in numerosi contesti applicativi, inclusa l'interpretazione di enunciati nei sistemi di dialogo, in particolare nelle applicazioni di e-commerce, dove decine di migliaia di descrizioni di prodotti per vari domini e lingue differenti devono essere analizzate. Proponiamo un modello neurale supervisionato che riesce a raggiungere lo 0.79 di F-measure, che consideriamo un risultato eccellente data la minima quantitá di supervisione richiesta e la alta complessitá linguistica del dominio.*

1 Introduction

Noun-compounds are nominal descriptions that hold implicit semantic relations between their constituents (Shwartz and Dagan, 2018). For instance, an *apple cake* is a cake made of apples. While in the literature there has been a large interest in interpreting noun-compounds by classifying them with a fixed set of ontological relations (Nakov and Hearst, 2013), in this paper we focus on automatic recognition of the head-noun in noun-compounds. We assume that in each noun-compound there is a noun which can be considered as the more informative, as it brings the most relevant information that allows the correct interpretation of the whole noun-compound. For instance, in the *apple cake* example, we consider *cake* as the head-noun, because it brings more information than *apple* about the kind of food the compound describes (i.e. a dessert), its ingredients (i.e. likely, flour, milk and eggs), and the typical amount a person may eat (i.e. likely, a slice). While in simple noun-compounds the head-noun usually corresponds to the syntactic head of the compound, this is not the case for complex compounds, where the head-noun can occur in different positions of the compound, making its identification challenging. As an example, in the Italian food description *filetto di vitellone senza grasso visibile*, there are three nouns (i.e. *filetto*, *vitellone* and *grasso*) which are candidates to be the head-noun of the compound.

There are a number of tasks and application domains where identifying noun-compound head-nouns is relevant. A rather general context is ontology population (Buitelaar et al., 2005), where entity names automatically recognized in text are confronted against entity names already present in an ontology, and have to be appropriately matched in the ontology taxonomy. Our specific application interest is conversational agents for the e-commerce domain. Particularly, understanding names of products (e.g. food, furniture, clothes, digital equipment) as expressed by users in different languages, requires the capacity to distinguish

the main element in a product name (e.g. a *table* in *I am looking for a three legs table with white marble top*), in order to match them against vendor catalogues and to provide a meaningful dialogue with the user. The task is made much more challenging by the general lack of annotated data, so that fully supervised approaches are simply not feasible. Along this perspective, the long term goal of our work is to develop unsupervised techniques that can identify head-nouns in complex noun-compounds by learning properties on the base of the noun-compounds included in, possibly large, gazetteers, regardless of the domain and language in which they are described.

In this paper we propose a supervised approach based on a neural sequence-to-sequence model (Lample et al., 2016) augmented with noun-compound structural features (Guerini et al., 2018). This model identifies the more informative token(s) in the noun-compound, that are finally tagged as the head-noun. We run experiments on Italian food names, and show that, although the domain is very complex, results are promising.

The paper is structured as follow: we first define noun-compound head-noun identification, with specific reference to complex noun-compound (Section 2). Then we introduce the neural model we have implemented (Section 3), and finally the experimental setting and the results we have obtained (Section 4).

2 Food Compound-Nouns

In this Section we focus on Italian compound-nouns referring to food, the domain on which we run our experiments. Similar considerations and same methodology can be applied to compound-nouns in different domains and languages.

There is a very high variety of food compound-nouns, describing various aspects of food, including: simple food names, like *mortadella di fegato*, *pesce*, *gin and tonic*, *aglio fresco*; recipes mentioning their ingredients, like *scaloppine al limone*, *spaghetti al nero*, *passato di pollo*, *decotto di carciofo*; recipes focusing on preparation style, like *mandorle delle tre dame*, *cavolfiore alla napoletana*; food names focusing on visual or shape properties, like *filetto di vitellone senza grasso visibile*, *palline di formaggio fritte*; food descriptions containing a course name, like *antipasto di capesante*, *dessert di mascarpone*; food using fantasy names, like *frappé capriccioso*, or *insalata arlecchino*; food including proper names or brands, like *saint-honoré*, *tagliatelle Matilde*, *formaggio bel paese*; food names focusing on cooking modalities, like *pane fatto in casa*, or *peperoni fritti*; and focusing on alimentary properties, like *ragù di carne dietetico*, or *sangria analcolica*.

We assume that the head-noun of a food description is the more informative noun in the noun-compound, i.e. the noun that better allows to answer questions about properties of the food being described by the noun-compound. We consider the following four property related questions, in order of relevance:

1. What *food category* (e.g. meat, vegetable, cake, soup, pasta, fish, liquid, salad, etc.) is described by the noun-compound?
2. What *course* (e.g. main, appetizer, side dish, dessert, etc.) is described by the noun-compound?
3. Which is the *main ingredient* (in term of quantity) described by the noun-compound?
4. Which could be the overall *quantity* (expressed in grams) of food described by the noun-compound?

Although our approach does not require any domain knowledge, for the purpose of human annotation and evaluation it is useful to assume a simple ontology for food, where we define the properties used for judging head-nouns and the set of possible values for each property. Table 1 reports the food ontology at the base of our work.

Property	Values
Food category	meat, vegetable, cake, soup, pasta, fish, liquid, salad...
Course	main, first, second, appetizer, side , dessert...
Main ingredient	<simple food>
quantity	<grams>

Table 1: Food Ontology.

A good head-noun should be as much informative as possible about the noun-compound properties, or, in other terms, it should allow to infer as much as possible answers to questions 1-4. Answers to such questions are in most of the cases

graduated and probabilistic, as a noun-compound contains just a fraction of the knowledge needed to answer them. For instance, given question 1) for the food noun-compound *insalata noci e formaggio* should be posed in the following way: knowing that *formaggio* is part of a food description, which is the probability that the overall description correctly refers to a food of category *salad*? When the probability is very low, we assume a "no guess" value for the answer.

The core procedure for human annotations considers each content word in a food description, fills in the values of the four attributes, and then select the noun with the best guesses. Below some examples (in black the selected head of the food description):

- ***insalata*** *noci e formaggio*: because *insalata* is a better predictor of the food category than *formaggio* or *noci*.
- *involtini di* ***peperoni***: because *peperoni* is a better predictor of food category (i.e. vegetable) and of the main ingredient than *involtini*.
- ***budino*** *al cioccolato fondente*: because *budino* is a good predictor of food category (i.e. dessert) and a better predictor than *cioccolato* of the main ingredient (i.e. milk) of the noun-compound.

2.1 Task and Data Set

Given a food noun-compound, the task we address is to predict its head-noun, labelling one or more consecutive tokens in the food description. We assume that a head is always present, even in case it is poorly informative.

Two annotators were selected to annotate a data set of 436 food names, collected from recipe books, with their head-noun. The inter annotator agreement, computed at the token level, is Cohen's kappa: 0.91, which is considered very high.

In table 2 we give an overview of the data set of food-description head (FDH) we created focusing on two main orthogonal characteristics: whether the head-noun is comprised of a single token or of a multi-token, and whether the head-noun corresponds to the beginning of the food description or not. As can be seen, the vast majority of head-nouns is either made of a single token (almost 90% of cases), or starts at the beginning of the entity name (almost 80% of cases). The combination of the two accounts for roughly 70% of the cases. From the point of view of predicting the head-noun of a food name, easier cases are given by single token in first position, while harder cases are given by multi-token head inside the food name.

	FDH type		
Position	Single token	Multi token	Total
1^{st} token	72.48	9.17	81.65
Not 1^{st} token	17.89	0.46	18.35
Total	90.37	9.63	

Table 2: Coverage on the data set of head-noun characteristics (in %): either single token or multi-token and whether appearing at the beginning of the food description or not.

3 Model

The architecture we use to recognize head-nouns is based on a bidirectional LSTM (Long Short Term Memory) network (Graves and Schmidhuber, 2005), similar to the one presented in (Lample et al., 2016). We briefly describe the LSTM model used in the approach and proceed with the implementation details.

3.1 LSTM

Recurrent Neural Network (RNN) is a class of artificial neural network that resemble a chain of repeating modules to efficiently model sequential data (Mikolov et al., 2010). They take sequential data $(x_1, x_2, ...x_n)$ as input and provide a representation $(h_1, h_2, ...h_n)$ which captures the information at every time step in the input. Formally,

$$h_t = f(Ux_t + Wh_{t-1})$$

where x_t is the input at time t, U is the embedding matrix, f is a non-linear operation (such as sigmoid, tanh or ReLU) and W is the parameter of RNN learned during training.

The hidden state h_t of the network at time t captures only the left context of the sequence for the input at time t. The right context for the input at time t can be captured by performing the same operation in the negative time direction. The input can be represented by both its left context $\overrightarrow{h_t}$ and right context $\overleftarrow{h_t}$ as $h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}]$. Similarly, the representation of the completed sentence is given by $h_T = [\overrightarrow{h_T}; \overleftarrow{h_0}]$. Such processing of the input in both forward and backward time-step is known as bidirectional RNN. Though a vanilla RNN is good

at modelling sequential data, it struggles to capture the long-term dependencies in the sequence. Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) is a special kind of RNN that is designed specifically to capture the long-term dependencies in sequential data. They compute the the hidden state h_t as follows,

$$
\begin{aligned}
i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i \\
f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f \\
\tilde{C}_t &= tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\
C_t &= f_t * C_{(t-1)} + i_t * \tilde{C}_t \\
o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o \\
h_t &= o_t * tanh(C_t)
\end{aligned}
$$

where x_t is the embedding for input at time t; i_t, f_t, o_t are the input, forget and output gates, respectively.

3.2 Implementation

The task of head-noun identification aims to predict a sequence of tags $y = \{y_1, y_2, .., y_n\}$ given an input sequence $X = \{x_1, x_2, ..x_n\}$. The system is modeled as a sequence labelling task and consists of three main steps: i) *word embedding*: each word in the sequence is embedded to a higher dimension; ii) *Input encoder*: encoding the sequence of embeddings; iii) *Classification*: labelling the sequence.

Word embeddings. Each word in the input sequence is represented by a vector of d-dimensions that captures the syntactic and semantic information of the word. The representation is carried by a word embedding matrix $E \in \mathbb{R}^{d \times |v|}$ where $|v|$ is the input vocabulary size. In addition to this, the model combines a character embedding that is learned during training using a Bi-LSTM network to deal with out of vocabulary terms and possible misspellings (Ling et al., 2015).

To represent the core structure of a complex noun-compound, we also use the following handcrafted features of a head-noun candidate token (Guerini et al., 2018): (i) the actual position of the token within the compound name; (ii) the length of the candidate token; (iii) the frequency of the token in the gazetteer; (iv) the average length of the noun-compounds in the gazetteer containing the token; (v) the average position of the token in the noun-compound it appears in; (vi) the bigram probability with reference to the previous token in the noun-compound; (vii) if the token can be an noun-compound; (viii) the ratio of the time the token is the first token in a noun-compound; (ix) the ratio of the time the token is the last token in a noun-compound. These handcrafted features for each word are extracted from a large corpus of Italian food names reported in (Guerini et al., 2018).

The concatenation of word embedding, final states of bidirectional character embeddings network, and hand crafted features is used as the word representation.

Input encoder. LSTM nodes are used to encode the input sequence of word embeddings. We employ a bidirectional LSTM (Bi-LSTM) to capture the context in both forward and backward timesteps. The hidden representation of a word at time t is given as,

$$h_t = [\overrightarrow{h}_t; \overleftarrow{h}_t]$$

Classification. The output layer receives the hidden representation from the Bi-LSTM and outputs a probability distribution over the possible tag sequences. Then, a conditional random field (CRF) layer (Lafferty et al., 2001) is used to model the dependency in labelling tags. The hidden representations from the Bi-LSTM are passed through a linear layer to obtain the score P_i for each word in the input sequence $X = \{x_1, x_2, .., x_n\}$. The score for each possible output tag sequence $\hat{y} \in \hat{\mathbf{Y}}$ is then obtained as follows,

$$Score(\hat{y}) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i}$$

where A is the transition matrix representing the transition scores from tag i to tag j. The probability of the tag sequence is then computed using a softmax operation as follows,

$$p(\hat{y}|X) = \frac{exp(Score(\hat{y}))}{\sum_{\tilde{y} \in \hat{\mathbf{Y}}} exp(Score(\tilde{y})}$$

The training is done by maximizing the log probability of the correct output tag sequence.

4 Experiments and Results

4.1 Setup

The dimension of character embedding is set to 30 and embeddings are learned using 50 hidden units

in each direction. For the word embeddings, as learning this level of representation with a small dataset is highly inefficient, we decided to use pre-trained embeddings trained using skip-gram (Mikolov et al., 2013) on the Italian corpus of Wikipedia. The input encoder consists of 120 hidden units in each direction with a dropout (E. Hinton et al., 2012) of 0.5 applied between the Bi-LSTM layer and the output layer.

4.2 Baselines

To compare the performance of the proposed approach, we provide two baselines: i) *1st token*, where the 1st token of a noun-compound is chosen as its head-noun; ii) *Spacy*[1], where the root token of the dependency tree for the noun-compound is chosen as its head-noun.

1st token. This baseline implicitly accounts for a number of linguistic behaviours of head-nouns in Italian language: (a) avoids stop words as head-nouns, as they do not occur at the first position of a noun-compound; (b) avoids adjectives as head-nouns, as they usually occur after the noun they modify; (c) captures the syntactic head of the noun-compound, which, in Italian is likely to be the first noun in a Noun Phrase; as already seen in Table 2. Summing up, the first-token baseline captures relevant linguistic behaviours, and is a strong competitor of our neural model, as in more than 80% of the entries in our dataset the first token belongs to head-noun of the noun-compound.

Spacy. This is a widely known open-source library for natural language processing and include a syntactic dependency parser. Given an input sequence, based on the result returned by the dependency parser, the root of the sequence is chosen to be the head-noun. We used the statistical model *it_core_news_sm*[2] released by Spacy for Italian language.

4.3 Evaluation metric

The performance of the models are evaluated using F1 score as in CoNLL-2003 NER evaluation (Sang and Meulder, 2003), which is a standard for evaluating sequence tagging tasks.

4.4 Results

The results for the FDH dataset are shown in Table 3. The baselines *1st token* and *Spacy* achieve a performance of 70.27 of 62.67 respectively. In particular, the performance of syntactic dependency parser from Spacy reiterates the difference between the semantic and syntactic head. The results are shown by incremental features, for the proposed approach. The models reported without CRF, are trained using a softmax function as output layer to predict the tag. We can notice from the results that using only the pre-trained embeddings, the network suffers from a poor recall and fails to achieve even the baseline performance. However, using either character embedding or the hand-crafted features, improves the performance of the model on par with the baseline. Since the single token head-noun in FDH dataset is very high (as shown in table 2), learning the multi token head-nouns and the dependency of tags is a challenge. However, introducing the CRF layer to jointly predict the sequence of tags in combination with the hand crafted features, enables us to predict multi-token heads and improve the performance of the model to 78.09. Finally, the character embeddings learned during training helps to improves the recall further, reaching a F1 score of 79.58.

	Accuracy	Precision	Recall	F1
Baselines				
1st token	83.74	70.29	70.24	70.27
Spacy	78.47	62.70	62.67	62.67
Bi-LSTM				
a) word_emb	84.06	74.10	65.18	69.28
b) a + hc_feat	85.17	75.76	66.50	70.76
c) a + char_emb	85.21	76.24	66.28	70.79
d) b + CRF	88.07	78.57	77.67	78.09
d) d + char_emb	**88.59**	**80.58**	**78.62**	**79.58**

Table 3: Experimental results on FDH dataset.

5 Conclusion and Future Work

We have addressed head-noun identification in complex noun-compounds, a task of high relevancy in utterance interpretation for dialogue systems. We proposed a neural model, and experiments on Italian food noun-compounds show that the model is able to outperform strong baselines even with a small amount of data. For the future we plan to extend our investigation to other domain and languages.

References

Paul Buitelaar, Philipp Cimiano, and Bernardo Magnini. 2005. *Ontology Learning from Text:*

[1] https://spacy.io/
[2] https://spacy.io/models/it

Methods, Evaluation and Applications, volume 123 of *Frontiers in Artificial Intelligence and Applications Series*. IOS Press, Amsterdam, 7.

Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. arXiv, 07.

A. Graves and J. Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm networks. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 4, pages 2047–2052 vol. 4, July.

Marco Guerini, Simone Magnolini, Vevake Balaraman, and Bernardo Magnini. 2018. Toward zero-shot entity recognition in task-oriented conversational agents. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 317–326, Melbourne, Australia, July.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

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.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *CoRR*, abs/1603.01360.

Wang Ling, Chris Dyer, Alan W. Black, Isabel Trancoso, Ramon Fermandez, Silvio Amir, Luís Marujo, and Tiago Luís. 2015. Finding function in form: Compositional character models for open vocabulary word representation. In *EMNLP*.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH 2010)*, volume 2010, pages 1045–1048. International Speech Communication Association.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Preslav Nakov and Marti A. Hearst. 2013. Semantic interpretation of noun compounds using verbal and other paraphrases. *TSLP*, 10(3):13:1–13:51.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *CoRR*, cs.CL/0306050.

Vered Shwartz and Ido Dagan. 2018. Paraphrase to explicate: Revealing implicit noun-compound relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1200–1211. Association for Computational Linguistics.

La sentiment analysis come strumento di studio del parlato emozionale?

Paolo Mairano
University of Lille, France
paolo.mairano@univ-lille.fr

Enrico Zovato
University of Turin, Italy
ezovato@gmail.com

Vito Quinci
University of Turin, Italy
vito.quinci540@edu.unito.it

Abstract

Italiano. Vari studi in letteratura hanno dimostrato che il parlato emozionale è caratterizzato da vari indici acustici. Tuttavia, tali studi hanno quasi sempre utilizzato parlato recitato, ignorando il parlato elicitato in maniera ecologica a causa della difficoltà nel reperire adeguate produzioni emozionali. In questo contributo, esploriamo la possibilità di utilizzare la *sentiment analysis* per selezionare produzioni emozionali da corpora orali. Abbiamo utilizzato il corpus *LibriSpeech*, da cui abbiamo estratto valori di *sentiment analysis* a livello di frase e di parola, nonché vari indici acustici e spettrali associati al parlato emozionale. L'analisi della relazione tra i livelli acustico e testuale ha rivelato effetti significativi ma di portata ridotta. Questo ci fa pensare che tali due livelli (acustico e lessicale) tendano a essere relativamente indipendenti, rendendo inappropriato l'utilizzo di metriche testuali per la selezione di materiale acusticamente emozionale.

English. *Abundant literature has shown that emotional speech is characterized by various acoustic cues. However, most studies focused on sentences produced by actors, disregarding ecologically elicited speech due to difficulties in finding suitable emotional data. In this contribution we explore the possibility of using sentiment analysis for the selection of emotional chunks from speech corpora. We used the LibriSpeech corpus and extracted sentiment analysis scores at word and sentence levels, as well as several acoustic and spectral parameters of emotional voice. The analysis of the relation between textual and acoustic indices revealed significant but small effects. This suggests that these two levels tend to be fairly independent, making it improper to use sentiment analysis for the selection of acoustically emotional speech.*

1 Introduzione

L'espressione delle emozioni può avvenire attraverso diversi componenti a vari livelli linguistici (Reilly & Seibert, 2003): lessicale (verbi modali, elementi rafforzativi, attenuativi, o valutativi), sintattico (es. le proposizioni relative possono commentare azioni e comportamenti), acustico (prosodia, qualità della voce), e paralinguistico (espressioni del viso, gesti). I framework tradizionali per l'analisi delle emozioni sono basati su categorie (Ekman, 2000) o su dimensioni (Russell, 1980). I primi distinguono vari stati emozionali (rabbia, gioia, paura, tristezza, etc.), mentre i secondi tendono a definire le emozioni come coordinate in uno spazio multidimensionale, in cui ogni dimensione rappresenta una proprietà di uno stato emozionale. Tra i numerosi framework esistenti, Russell (1980) ipotizza due dimensioni: *valence* (valenza, positiva vs. negativa) e *arousal* (attivazione, alta vs. bassa). La classificazione degli stati emozionali tramite indizi linguistici si è rivelata un compito arduo tanto nei framework categoriali quanto in quelli dimensionali, e l'interazione dei vari livelli linguistici complica ulteriormente la situazione: non è ancora chiaro se la componente lessicale / sintattica debba essere considerata come dipendente o complementare alla componente acustica.

Nonostante tali problemi, molti studi hanno analizzato il parlato emozionale con l'obiettivo di individuare i correlati acustici specifici dei vari stati emozionali. Alcuni studi hanno dimostrato che variazioni sistematiche della frequenza fondamentale (sia in termini di *pitch range*, sia in termini di *pitch* medio) accompagnano realizzazioni di parlato con valenza positiva (Burkhardt & Sendlmeier, 2000). Ma anche altri parametri prosodici sembrano avere un ruolo importante nella comunicazione delle emozioni: sono stati

riscontrati effetti dell'intensità e della velocità d'eloquio (Johnstone & Scherer, 2000); infatti, varie misure acustiche (deviazione standard della frequenza fondamentale, energia media, durata dei periodi, *spectral-dropoff*, etc.) sono state usate per predire i giudizi di parlanti madrelingua (Banse & Scherer, 1996) e vari altri parametri sono stati usati in altri studi (cf. Schröder et al., 2001, e Audibert, Aubergé & Rilliard, 2005).

Tuttavia, uno dei limiti di questi studi riguarda l'affidabilità dei dati: data la difficoltà di elicitare parlato emozionale controllato, gran parte degli studi utilizza registrazioni di parlato recitato, che spesso risulta stereotipato o esagerato (Scherer, 2003). In questo contributo, verifichiamo se la *sentiment analysis* (d'ora in poi: SA) possa essere d'aiuto in questo senso. La SA, ovvero lo studio delle opinioni, sentimenti, recensioni delle persone in forma testuale (Liu, 2003), è un settore NLP in rapida crescita, grazie anche all'ampio ventaglio di applicazioni, quali la classificazione di email (Mohammad & Yang, 2011), romanzi (Mohammad, 2011), recensioni cinematografiche (Sadikov, 2009), recensioni di articoli o servizi acquistati (McGlohon, Glance & Reiter, 2010). I sistemi di SA vanno da metodi a regole relativamente semplici, fino a tecniche avanzate di *deep learning* - vedi Liu (2012) per una rassegna.

In questo studio, verifichiamo la relazione tra i valori di SA e le caratteristiche acustiche del parlato letto elicitato in maniera naturale, estrapolate da audiolibri. Il fine ultimo è quello di estendere l'analisi a dati di parlato spontaneo; tuttavia, dati i numerosi problemi che questo tipo di parlato comporta, abbiamo preferito iniziare da dati di parlato letto in cui le emozioni non fossero state elicitate esplicitamente. Per misurare il grado di emozione espresso dal testo degli audiolibri, sono stati utilizzati *SentiWordNet* (Baccianella, Esuli & Sebastiani, 2010) e *Vader* (Gilbert & Hutto, 2014), che operano principalmente a livello lessicale. Sul piano acustico, abbiamo estratto vari indici (per lo più prosodici) descritti in letteratura. Un'analisi simile a questa, che studia la correlazione tra SA e parametri acustici, è stata condotta da Charfuelan & Schröder (2012) su dati di un solo speaker e di un solo audiolibro. Qui estendiamo l'analisi a 251 audiolibri letti da speaker diversi, nella speranza che i risultati abbiano sia rilevanza teorica (studio dell'interazione tra livello lessicale e acustico nel parlato emozionale), sia un risvolto pratico (utilizzo della SA per la selezione di parlato emozionale non recitato).

2 Dati e metodologia

2.1 Corpus

Per studiare la correlazione tra i valori della SA e le caratteristiche acustiche del parlato emozionale, abbiamo utilizzato *LibriSpeech* (Panayotov et al., 2015), un corpus open-source contenente circa 1000 ore di parlato in inglese. I dati di *LibriSpeech* provengono a loro volta dal progetto *LibriVox* (una collezione di audiolibri di dominio pubblico, disponibili su librivox.org), e i testi sono stati segmentati e allineati automaticamente dagli autori del corpus. Ai fini di questo studio, abbiamo limitato l'analisi alla sezione *train-clean-100* del corpus (contenente 100 ore di parlato corretto e pulito, originariamente concepito come *training set* per sistemi ASR), che include i dati di 251 audiolibri. I lettori sono un mix di professionisti e non-professionisti di sesso maschile e femminile (l'età non è riportata). L'elenco dei testi registrati (consultabile sul sito web di LibriVox) include principalmente opere letterarie britanniche e americane, antiche e moderne.

Tutto il materiale è stato trascritto foneticamente con il front-end del sistema TTS *Vocalizer* di *Nuance Communications*, secondo il modello di General American. Le trascrizioni sono poi state allineate al segnale acustico e infine convertite in formato *TextGrid* per essere utilizzate con *Praat* (Boersma & Weenink, 2018).

2.2 Metriche di *sentiment analysis*

I valori di SA sono stati estratti dal testo di ciascuna frase usando strumenti open-source, quali *Vader* (Gilbert & Hutto, 2014) e *SentiWordNet* (Baccianella et al., 2010), entrambi disponibili nella libreria NLTK di Python. Si tratta di strumenti classici nella letteratura sulla SA e relativamente semplici dal punto di vista dell'utilizzo e dell'implementazione (trattandosi di sistemi a regole). In futuro, l'analisi potrebbe essere estesa utilizzando strumenti più complessi e sofisticati, come i modlui di SA dei progetti OpeNER (http://www.opener-project.eu/) e StanfordNLP (https://nlp.stanford.edu/).

Vader fornisce tre valori: (a) un punteggio di polarità positiva compreso tra 0 e 1 (*Vader_comp*), (b) un punteggio di polarità negativa compreso tra 0 e 1, (c) un punteggio

derivato dagli altri due compreso tra -1 (negativo) e +1 (positivo). Questi valori sono ricavati grazie a un sistema a regole, basato sul lessico *Vader*, nel quale le parole sono associate a un punteggio di polarità ottenuto dalle valutazioni di 13 madrelingua. *SentiWordNet* adotta un approccio leggermente diverso: le parole nel suo lessico sono associate a punteggi di polarità positiva o negativa procurati tramite un'analisi quantitativa di ogni synset (vedi Baccianella et al., 2010, per maggiori dettagli).

I valori di *Vader_comp* sono stati valutati sulla base di un sottoinsieme di 1000 frasi annotate manualmente da uno degli autori (prendendo frasi isolate, quindi senza informazioni sul contesto o sul co-testo), ottenendo un'accuratezza pari al 72%.

2.3 Indici acustici del parlato emozionale

Sebbene la maggior parte degli studi si concentrino sui parametri acustici a livello di frase, noi abbiamo applicato l'analisi anche a livello di parola, sulla base dell'ipotesi che le parole con carica emozionale possano essere caratterizzate da specifici indici acustici (Tsiakoulis et al., 2016).

Per l'analisi a livello di frase, gli indici acustici sono stati estratti per ogni frase. Per l'analisi a livello di parola, invece, gli indici acustici sono stati estratti dalla vocale accentata delle parole non funzionali al fine di controllare le differenze spettrali dei vari fonemi vocalici (il fonema vocalico è stato incluso come fattore nell'analisi statistica). I seguenti indici acustici sono stati estratti tramite *Praat*: F0 mean (frequenza fondamentale media in semitoni), F0 stdev (in semitoni), F0 range (0.05-0.95), F0 max (0.95), F0 min (0.05), shimmer, jitter, Hammarberg index (HAM, differenza tra il massimo di energia nelle bande 0-2 kHz e 2-5kHz, cf. Hammarberg et al., 1980), Do1000 (riduzione di energia spettrale oltre 1000 Hz), Pe1000 (energia relativa a frequenze oltre 1000 Hz vs energia sotto i 1000 Hz, cf. Scherer, 1989, e Drioli et al., 2003). I valori di F0 sono stati estratti tramite il metodo di autocorrelazione di *Praat* (con i parametri di default) secondo una procedura in 2 fasi: in una prima fase, l'estrazione è stata fatta con un range fisso 75-400 Hz; l'intervallo interquartile (IQR) è stato calcolato sui valori così ottenuti, e una seconda estrazione è stata realizzata nel range tra +50% e -25% dall'IQR.

Inoltre, per l'analisi a livello di frase abbiamo estratto la durata totale in ms dal primo all'ultimo fonema (DUR), *speech rate* (SR, numero di fonemi diviso la durata complessiva incluse le pause), *articulation rate* (AR, senza le pause), *pause/speech ratio* (PSR).

Tutti i parametri acustici estratti sono stati trasformati in *z-scores* per ogni speaker, nel tentativo di normalizzare le differenze tra speakers. Le frasi contenenti meno di 3 secondi di parlato sono state escluse dall'analisi. Per ogni parametro acustico, i valori che si scostavano >2.5 deviazioni standard dalla media sono stati esclusi come probabili errori di detezione.

3 Risultati

3.1 Analisi a livello di frase

I dati sono stati analizzati su *R* tramite modelli a effetti misti con la libreria *lme4* (Bates et al., 2014) per valutare la relazione tra valori di SA e parametri acustici. In una prima analisi, abbiamo costruito dei modelli per valutare l'effetto di *Vader_comp* (che prendiamo come indicativo di valenza) su ogni indice acustico separatamente, includendo sempre il fattore speaker come effetto aleatorio, es.: *F0_range ~ Vader_comp + (1 | speaker)*. Questa prima analisi ha rivelato che il valore di *Vader_comp* ha un effetto significativo sui valori di F0, in particolare F0 max, F0 range, F0 mean e F0 stdev (v. tabella 1).

Modello	**p val.**
F0 min ~ Vader_comp + (1|speaker)	*ns*
F0 max ~ Vader_comp + (1|speaker)	***
F0 range ~ Vader_comp + (1|speaker)	***
F0 mean ~ Vader_comp + (1|speaker)	***
F0 stdev ~ Vader_comp + (1|speaker)	***
AR ~ Vader_comp + (1|speaker)	*ns*
PSR ~ Vader_comp + (1|speaker)	*p = .05*
Shimmer ~ Vader_comp + (1|speaker)	*ns*
Jitter ~ Vader_comp + (1|speaker)	*ns*
HNR ~ Vader_comp + (1|speaker)	*ns*
Do1000 ~ Vader_comp + (1|speaker)	*ns*
Pe1000 ~ Vader_comp + (1|speaker)	*ns*
HAM ~ Vader_comp + (1|speaker)	*ns*

Tabella 1. Effetto di *Vader_comp* sui valori acustici.

L'effetto di *Vader_comp* non è risultato significativo per la predizione degli indici di ritmo e durata. Quindi abbiamo voluto verificare se questi parametri si correlino con l'intensità di attivazione, piuttosto che con la valenza. Abbiamo quindi valutato modelli separati per frasi negative (*Vader_comp* < 0) vs positive

(*Vader_comp* > 0). Tali modelli hanno mostrato che il valore *Vader* di positività (range:0-1) ha un effetto significativo non solo sugli indici di F0, ma anche su AR, PSR, shimmer, HNR, Do1000, Pe1000 e HAM. Analogamente, il valore *Vader* di negatività (range:0-1) ha un effetto significativo per gli indici di F0, nonché su AR e shimmer (v. tabella 2).

Effetto	Vader>0	Vader<0
F0 min	***	***
F0 max	*ns*	***
F0 range	*	**
F0 mean	**	*ns*
F0 stdev	*	*
AR	***	***
PSR	*	*ns*
Shimmer	***	**
Jitter	*ns*	*ns*
HNR	*	*ns*
Do1000	*	*ns*
Pe1000	*	*ns*
HAM	*	*ns*

Tabella 2. Effetto di *Vader_pos* e *Vader_neg* sui valori acustici.

Questi risultati sembrano quindi suggerire che gli indici di F0 siano influenzati dalla valenza della frase, mentre gli indici ritmici e spettrali si correlano con l'intensità di positività o negatività della frase. Tuttavia, la parte di varianza spiegata dai vari modelli rimane bassa, con ad esempio R^2 = 0.01 per il modello che predice AR.

Infine, abbiamo costruito un modello a effetti misti per predire *Vader_comp* a partire dagli indici acustici, includendo il fattore 'speaker' come effetto aleatorio. Dopo l'eliminazione degli effetti non significativi, abbiamo ottenuto R^2 = 0.06 per la contribuzione cumulativa di tutti gli indici acustici significativi. Considerando separatamente le frasi con valori positive e negativi (cercando quindi di predire i valori *Vader* di positività e negatività sulla base degli indici acustici), R^2 sale a 0.09 per il modello che predice i valori *Vader* di positività, e a 0.12 per il modello che predice i valori *Vader* di negatività.

3.2 Analisi a livello di parola

Analogamente a quanto fatto a livello di parola, in una prima analisi abbiamo costruito dei modelli a effetti misti per valutare la relazione tra valori di SA e ognuno dei parametri acustici separatamente. Come variabile *predictor* abbiamo utilizzato il valore di valenza per ogni parola nel lessico di *Vader*, e abbiamo incluso il fattore 'speaker' come effetto aleatorio. Inoltre, per i parametri spettrali HNR, Do1000, Pe1000 e HAM abbiamo incluso il fattore 'fonema' come effetto aleatorio, poiché tali parametri variano in funzione delle diverse vocali. Come per l'analisi a livello di frase, i modelli ci dicono che il valore di valenza di *Vader* ha un effetto significativo sugli indici F0 min, F0 range, F0 mean, F0 stdev, e questa volta anche shimmer e jitter (v. tabella 3).

Modello	p val.
F0 min ~ Vader + (1|speaker)	***
F0 max ~ Vader + (1|speaker)	*ns*
F0 range ~ Vader + (1|speaker)	***
F0 mean ~ Vader + (1|speaker)	***
F0 stdev ~ Vader + (1|speaker)	***
Shimmer ~ Vader + (1|speaker)	***
Jitter ~ Vader + (1|speaker)	**
HNR ~ Vader + (1|speaker)	*ns*
Do1000 ~ Vader + (1|speaker)	*ns*
Pe1000 ~ Vader + (1|speaker)	*ns*
HAM ~ Vader + (1|speaker)	*ns*

Tabella 3. Effetto di *Vader* sui valori acustici.

In una seconda analisi, come a livello di frase, abbiamo voluto verificare se i parametri acustici fossero correlati all'intensità di attivazione positiva o negativa della parola. Per far questo, abbiamo costruito altri modelli separati per parole con valenza positiva (*SentiWordNet pos value* > 0) in frasi positive (*Vader_comp* > 0) e per parole con valenza negativa (*SentiWordNet neg value* > 0) in frasi negative (*Vader_com* < 0). I modelli relativi a parole positive hanno rivelato un effetto significativo di *SentiWordNet pos value* su HNR, Do1000 e Pe1000, ma solo marginalmente significativi sugli indici di F0. I modelli relativi a parole negative in frasi negative hanno rivelato un effetto significativo di *SentiWordNet neg value* su HNR, Do1000, Pe1000, e HAM (v. tabella 4).

Nell'analisi a livello di frase, la parte di varianza spiegata da questi modelli era più alta (R^2 = 0.4 per Do1000 e Pe1000) rispetto all'analisi a livello di parola; tuttavia, ciò è dovuto soprattutto all'integrazione del fattore 'fonema' all'interno dei modelli; la parte di varianza spiegata dai valori di *SentiWordNet* ha raggiunto solo 0.004 e 0.007 per Do1000 e Pe1000 rispettivamente.

Effetto	SentiWN>0	SentiWN<0
F0 min	*	*ns*
F0 max	*ns*	***
F0 range	*ns*	***
F0 mean	*	***
F0 stdev	*ns*	***
Shimmer	*ns*	**
Jitter	*ns*	***
HNR	***	***
Do1000	***	***
Pe1000	***	***
HAM	*ns*	*ns*

Tabella 4. Effetto di *SentiWordNet_pos* e *SentiWordNet_neg* sui valori acustici.

4 Conclusioni

La correlazione tra indici lessicali e acustici del parlato letto emozionale sembra essere significativa, ma di portata ridotta, sia a livello di parola, sia a livello di frase. Gli indici di F0 sembrano essere influenzati dalla valenza della frase e della parola, ma la parte di varianza spiegata rimane ridotta. Tali risultati confermano ed estendono quanto riportato da Charfuelan & Schröder (2012) su dati di un solo audiolibro, in cui erano state osservate correlazioni moderate per indici di F0 ed energia.

I dati mostrano una grande quantità di variabilità inter-speaker: risulta evidente che i locutori utilizzano diversi indici acustici per esprimere stati emozionali. Inoltre, un limite della nostra analisi risiede nell'utilizzo (inevitabile, data la mole di dati analizzati) di trascrizioni e annotazione automatiche, i cui errori causano senza dubbio un certo tasso di rumore nei dati, riducendo le relazioni osservabili tra le diverse variabili studiate. Infine, l'assenza di puntuazione nel corpus *LibriSpeech* rende impossibile (o molto complesso) differenziare tra discorso indiretto e diretto, nel quale ci si potrebbe aspettare un parlato più prettamente emozionale. Per il futuro, simili ipotesi potranno essere verificate su corpora più recenti costruiti con fini più specifici e adatti, come SynPaFlex (Sini et al., 2008).

Per concludere, riprendiamo il tema dell'interazione tra i vari livelli linguistici per l'espressione delle emozioni nel parlato. I risultati del nostro studio suggeriscono che i vari livelli linguistici analizzati (lessicale e acustico) sono relativamente slegati uno dall'altro per l'espressione delle emozioni. Questo significa che, per una determinata frase, i locutori hanno tendenza ad affidare l'espressione dello stato emozionale a uno solo dei due livelli analizzati. Questo può essere vero soprattutto per il parlato letto, in cui il locutore non è coinvolto direttamente, soprattutto nel caso del narratore di un audiolibro. Dunque, l'utilizzo della SA per lo studio del parlato emozionale appare non del tutto appropriato per selezionare materiale emozionalmente marcato, in quanto si baserebbe sull'assunzione che gli indici lessicali e acustici di emozionalità vadano di pari passo e tendano a co-occorrere. Tuttavia, rimane da esplorare la correlazione tra variabili lessicali e acustiche per altri tipi di parlato, in particolar modo per il parlato spontaneo – in cui i locutori siano più direttamente coinvolti rispetto al contenuto semantico.

Bibliografia

Audibert N., Aubergé V., Rilliard A. 2005. The prosodic dimensions of emotion in speech: the relative weights of parameters. *Proc. of* the *Ninth European Conference on Speech Communication and Technology*, 4-8 September 2005, Lisbon, Portugal.

Baccianella S., Esuli A., Sebastiani F. 2010. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *Proc. of LREC*, 17-23 May, Valletta, Malta, pp. 2200-2204.

Banse R., Scherer K.R. 1996. Acoustic profiles in vocal emotion expression, *Journal of personality and social psychology*, vol. 70, no. 3, 614–636.

Bates D., Maechler M., Bolker B., Walker S. 2014. Fitting Linear Mixed-Effects Models Using lme4, *Journal of Statistical Software*, vol. 67, no. 1, 1-48.

Boersma P., Weenink D.. 2018. *Praat: doing phonetics by computer* [Computer program]. Version 6.0.37, retrieved 3 February 2018 from http://www.praat.org/

Burkhardt F., Sendlmeier W.F. 2000. Verification of acoustical correlates of emotional speech using formant-synthesis. In *SpeechEmotion-2000*, pp. 151-156.

Charfuelan M., Schröder M. 2012. Correlation analysis of sentiment analysis scores and acoustic features in audiobook narratives. In *Proc. of the 4th International Workshop on*

Corpora for Research on Emotion Sentiment & Social Signals (ES3), 26 May 2012, Istanbul, Turkey, pp. 99-103.

Drioli C., Tisato G., Cosi P., Tesser F. 2003. Emotions and voice quality: experiments with sinusoidal modeling. In *Proc. of the Voice Quality: Functions Analysis and Synthesis (VOQUAL) Workshop*, 27-29 August, Geneva, Switzerland.

Ekman P. 2000. Basic Emotions. In T. Dalgleish and T. Power (eds.) *Handbook of Cognition and Emotion*, 39.6, London (UK), John Wiley & Sons, pp. 45-60.

Gilbert C.J., Hutto E. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proc. of the Eighth International Conference on Weblogs and Social Media* (ICWSM-14), 2-4 June, Ann Arbor MI, US.

Hammarberg B., Fritzell B., Gauffin J., Sundberg J., Wedin L. 1980. Perceptual and acoustic correlates of abnormal voice qualities, *Acta Otolaryngologica*, vol. 90, 441-451.

Johnstone T., Scherer K.R. 2000. Vocal communication of emotion. In M. Lewis and J. Haviland (eds.) *Handbook of emotions 2*, London-New York: The Guildford Press, pp. 220–235.

Liu B. 2012. Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, no.1, 1-167, 2012.

McGlohon M., Glance N., Reiter Z. 2010. Star quality: Aggregating reviews to rank products and merchants. In *Proc. of the International Conference on Weblogs and Social Media* (ICWSM-2010), 23-26 May, Washington DC, US.

Mohammad S. 2011. From Once Upon a Time to Happily Ever After: Tracking Emotions in Novels and Fairy Tales. In *Proc. of the ACL 2011 Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities* (LaTeCH), 24 June, Oregon, US.

Mohammad S., Yang T. 2011. Tracking Sentiment in Mail: How Genders Differ on Emotional Axes. In *Proc. of the ACL 2011 Workshop on Computational Approaches to Subjectivity and Sentiment Analysis* (WASSA-2011), 24 June, Alicante, Spain.

Panayotov V., Chen G., Povey D., Khudanpur S. 2015. Librispeech: an ASR corpus based on public domain audio books. In *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 19-24 April, Brisbane, Australia.

Reilly J., Seibert L. 2003. Language and emotion. In R.J. Davidson, K.R. Scherer and H.H. Goldsmith (eds.), *Handbook of affective sciences*, OUP, pp. 535–559.

Russell J.A. 1980. A circumplex model of affect, *Journal of personality and social psychology*, vol. 39, no.6, 1161-1178.

Sadikov E., Parameswaran A., Venetis P. 2009. Blogs as predictors of movie success. In *Proc of the Third International Conference on Weblogs and Social Media* (ICWSM-2009), 17-20 May, San Jose, CA, US.

Scherer K.R.. 1989. Vocal correlates of emotion. In A. Manstead and H. Wagner (eds.) *Handbook of psychophysiology: Emotion and social behavior*, London: Wiley, pp. 165-197.

Scherer K.R. 2002. Vocal communication of emotion: A review of research paradigms, *Speech Communication*, vol. 40, 227-256.

Schröder M., Cowie R., Douglas-Cowie E., Westerdijk M., Gielen S. 2001. Acoustic correlates of emotion dimensions in view of speech synthesis. In *Proc. of EUROSPEECH 2001 – Seventh European Conference on Speech Communication and Technology*, 3-7 September 2001, Aalborg, Denmark.

Sini A., Lolive D., Vidal G., Tahon M., E. Delais-Roussarie. 2008. SynPaFlex-Corpus: An Expressive French Audiobooks Corpus Dedicated to Expressive Speech Synthesis. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 7-12 May 2018, Miyazaki (Japan), pp. 4289-4296.

Tsiakoulis P., Raptis S., Karabetsos S., Chalamandaris A. 2016. Affective word ratings for concatenative text-to-speech synthesis. In *Proc. of the 20th Pan-Hellenic Conference on Informatics*, 10-12 November, Patras, Greece.

The iDAI.publication: extracting and linking information in the publications of the German Archaeological Institute (DAI)

Francesco Mambrini
Deutsches Archäologisches Institut
Podbielskiallee 69-71, Berlin
francesco.mambrini@dainst.de

Abstract

English. We present the results of our attempt to use NLP tools in order to identify named entities in the publications of the Deutsches Archäologisches Institute (DAI) and link the identified locations to entries in the iDAI.gazetteer. Our case study focuses on articles written in German and published in the journal *Chiron* between 1971 and 2014. We describe the annotation pipeline that starts from the digitized texts published in the new portal of the DAI. We evaluate the performances of geoparsing and NER and test an approach to improve the accuracy of the latter.

Italiano. *Il paper descrive i risultati dell'esperimento di applicazione di strumenti di NLP per annotare le Named Entities nelle pubblicazioni del Deutsches Archäologisches Institute (DAI) e collegare i toponimi identificati alle rispettive voci dell'*iDAI.gazetteer*. Il nostro studio si concentra sugli articoli in tedesco pubblicati nella rivista* Chiron *tra il 1974 e il 2014. Descriviamo la pipeline di annotazione impiegata per processare gli articoli disponibili nel nuovo portale per le pubblicazioni del DAI. Discutiamo i risultati della valutazione degli script di geoparsing e NER e, infine, proponiamo un approccio per migliorare l'accuratezza in quest'ultimo task.*

1 The iDAI.publications and the iDAI.world

The Deutsches Archäologisches Institute (German Archaeological Institute, henceforth DAI) is a German agency operating within the sphere of responsibility of the federal Foreign Office; the goal of the institue is to promote research in archaeological sciences and on ancient civilizations worldwide. Founded in Rome in 1829, the DAI has developed into a complex institution, with branches and offices located around the world. The Institute has participated in several projects, including missions of paramount importance like those in Olympia, Pergamon or Elephantine.

One of the most visible output of this activity is the amount of scientific publications produced by the DAI. The Institute currently publishes 14 international journals and 70 book series on different topics.[1] Since 2018, part of this collection is now accessible to the public on a new online portal named idai.publications for books and journals.[2] This ongoing initiative will not only enable researchers to have easier access to the published works; even more importantly, it will allow the Institute to integrate the data contained in articles and books (such as persons, places and archaeological sites, artifacts and monuments) into a network of all the other digital resources of the DAI.

All the digital collections of the DAI are indeed designed to operate within a network known as the idai.welt (or idai.world).[3] This network includes web collections such as "Arachne",[4] the database of archaeological monuments and artifacts of the DAI, and "Zenon",[5] the central bibliographic catalogue that serves all the libraries of the DAI offices around the world, but also compiles

[1] A list of journal is provided at: https://www.dainst.org/publikationen/zeitschriften/alphabetisch; for the list of book series: https://new.dainst.org/publikationen/reihen.

[2] See https://publications.dainst.org/journals/ and https://publications.dainst.org/books/.

[3] https://www.dainst.org/de/forschung/forschung-digital/idai.welt

[4] https://arachne.dainst.org/

[5] https://zenon.dainst.org/

some of the most comprehensive bibliographies in the areas of activity of the different branches.

The other cornerstone of the idai.world is represented by the layer of web-based services such as thesauri and controlled vocabularies. The idai.gazetteer,[6] in particular, connects names of locations with unique identifiers and coordinates; the gazetteer is intended to serve both as a controlled list of topnyms for DAI's services and to link the geographic data with other gazetteers. Unique identifiers defined in the idai.gazetteer are already used to connect places and entries in Zenon and Arachne. In this way, users of these services can already query monuments and artifacts in Arachne or books in Zenon that are linked to a specific place.

2 A pipeline for textual annotation

This network of references holds a great potential for the DAI publications. Places, persons, artifacts, monuments, and other entities of interest mentioned within the publications can be identified and linked to the concepts in the appropriate knowledge bases of the DAI. The linking of the different relevant entities would allow researchers not just to retrieve the texts that, independently from the language of the publication, make reference to certain concepts of interest, but also to study such epistemologically relevant questions as the variation in the patterns of locations cited in the studies across decades.

While the linking between entries in Zenon and Archne and the idai.gazetteer had been conducted manually, the volume and nature of the textual information to be processed in the publications encouraged us to turn to Natural Language Processing (NLP). We set up a pipeline for text annotation that aims to process the full texts of the publications, perform Named Entity Recognition (NER) to identify the mentions of the relevant entities, and finally link them to the appropriate entries in the idai.world.

We chose to build the first version of the pipeline around a series of open-source software that offer support for multiple languages and are widely used in the Digital Humanities (DH); at present, the annotation is limited to persons, places and organization, and only the linking of place-names to the idai.gazetteer is supported.

[6] https://gazetteer.dainst.org/

2.1 Preprocessing and NER

The pipeline is programmed in Python and takes advantages of modules of the NLTK platform for several task (Bird et al., 2009), like sentence- and word-tokenization.

The input of our annotation pipeline is, in the case of articles and books for which no other versions survive, the full text extracted from the PDF files of the articles.[7] The automatic recognition of the publication's main language is carried out by the Python library langid (Lui and Baldwin, 2011).

NER is performed using the Stanford Named Entity Recognizer (Finkel et al., 2005), which implements Conditional Random Field (CRF) sequence models. For a preliminary evaluation, we used pre-trained models for English, Spanish,[8] German (Faruqui and Padó, 2010), and Italian (Palmero Aprosio and Moretti, 2016). All these models are trained to recognize comparable classes of entities (persons, places, organizations and miscellaneous). We then chunked together the annotated tokens with a simple regular-expression chunker that takes consecutive, non-empty (O) tags together and labels them with the same label as the first token in the series.

Part-of-speech (POS) tagging, though not strictly necessary for NER and geoparsing, as the out-of-the-box models for Stanford NER do not require it, is also supported by our pipeline. TreeTagger (Schmid, 1999) was chosen since it offered a vast array of pre-trained models for many languages.

2.2 Geoparsing

The task of resolving place names by linking them to identifiers from a gazetteer is commonly referred to as "georparsing". The Edinburgh Geoparser[9] is a suite of tools that is often employed in DH (Grover et al., 2010; Alex, 2017) and allows users to preprocess texts, extract toponyms and resolve them by identifying the possible candidates in a gazetteer and scoring them. Users have the option to select between 4 gazetteers, and to set some parameters, like the coordinates of areas that will

[7] All the PDF files of the publications already include texts, so no Optical Character Recognition (OCR) is needed.

[8] Models for English and Spanish are available for download at https://stanfordnlp.github.io/CoreNLP/; for English we used the 4 Class model CoNLL 2003 English training set.

[9] http://groups.inf.ed.ac.uk/geoparser/documentation/v1.1/html/

be given preference while ranking the candidates. The scoring process makes use of some properties recorded for places in gazetteers (e.g. the type of location, such as inhabited place or archaeological site) and especially by comparing locations pairwise with all other places identified; preference is thus given to places that cluster together.

Although Edinburgh works only with English and the `idai.gazetteer` is not supported, the CLI software is built as a suite of scripts, so that the input of a process is the output of the preceding one. By knowing the script that performs a task and the input it expects, it is therefore possible to inject a pre-processed text into any given step, while most processes (like scoring) are language-agnostic. We integrated the ranking script of Edinburgh within our pipeline to score, for any location that we extracted with our own NER scripts, any list of possible candidates matched in the `idai.gazetteer`.

3 Testing and Improving The Pipeline: a case study

In this section we discuss the preliminary results obtained by running the pipeline described above on the complete series of one journal now available in the `idai.publications`. The results will serve as a baseline for future improvement.

3.1 *Chiron*: the data set

The first complete publication series that was added to the portal was *Chiron*, a journal published by the DAI's "Kommission für Alte Geschichte und Epigraphik" from 1970. Volumes from 1 to 44 (2014) are currently available,[10] for a total of 942 articles. The focus of the publication is in Graeco-Roman history and epigraphy; several articles contain lengthy quotations (or even full editions) of inscriptions in Greek or Latin.

Table 1 reports the total number of articles per language. As can be seen, quotations in Greek and Latin are sufficiently frequent and long to confuse the automatic recognition. In 39 cases, Latin or Greek were considered the main language of the publication. Luxembourgish (a West Germanic language) is also a clear mistake for German, also possibly prompted by lengthy quotations (Nollé and Wartner, 1987, for one likely case). The 44 volumes of the journal show an interesting distribution of languages, with German playing the most relevant role by far.[11]

[10] Readers are however requested to register an account.

Language	Nr. Articles	Auto rec.
German	645	580
English	211	222
French	59	55
Italian	17	15
Spanish	10	12
Luxembourgish	0	19
Greek and Lat.	0	39

Table 1: *Chiron*: number of article per language (actual count vs automatically recognized)

3.2 Evaluating the annotation

In this preliminary stage, we decided to focus on the 580 automatically identified German articles in order to evaluate the performances of our pipeline and to improve its accuracy.

We have manually corrected the NER annotation and geoparsing of 4 articles (Linke, 2009; Hammerstaedt, 2009; Sänger, 2010; Haensch and Mackensen, 2011), for a total of 36,159 words. The articles were selected so as to represent a broad scope of subjects (from papyrology, to social and religious history, to military archaeology) and geographic areas (North Africa, Asia Minor, Rome and Italy).

For the evaluation of our NER tools we adopted the same metrics (precision, recall and $F_{\beta=1}$ score) and methods of the CoNLL-2000 shared task (Tjong Kim Sang and Buchholz, 2000). Note, in particular, that the scores are calculated at the level of the phrase, not of the single tag. The evaluation of the geoparser is also based on the same principles, but instead of evaluating its performances on the automatically annotated texts, we re-ran the geoparser on the gold-standard and evaluated that output.

The scores reported in Table 2 are considerably below the state of the art in NER for German, as documented e.g. in the CoNLL 2003 shared task (Tjong Kim Sang and De Meulder, 2003). These results would very likely be considered insufficient or too noisy for the needs of researchers in the (Digital) Humanities.

[11] A word count on the automatically recognized languages confirms this conclusion: German has 7,394,004 words (60.48% of total), English 2,955,640, and French 899,888. Greek and Latin total 481,596 words; the other languages count between 193k and 148k words.

Entity	Precision	Recall	$F_{\beta=1}$
Person	73.21%	47.13%	57.34
Location	67.18%	34.56%	45.64
Organization	9.23%	35.71%	14.66
TOTAL	56.27%	43.22%	48.89

Table 2: NER: results of the first evaluation round; 1423 phrases; found: 1093; correct: 615

Modules for NER trained on general corpora do not seem to be suited to annotate texts that belong to such a specific domain with acceptable accuracy. The poor performances with organizations, in particular, point to some peculiarities of the archaeological literature in comparison to texts included in most general-use corpora: companies, firms and other institutions, which are frequent in the news, are rarely found in scholarly texts of our domain; the organization tag is more often reserved either to ancient institutions (like "the Roman Senate") or peoples and tribes ("the Aquitani") which are hardly represented in ordinary corpora.

Article	Precision	Recall	$F_{\beta=1}$
L09	76.53%	73.53%	75.00
H09	97.87%	95.83%	96.84
S10	72.66%	80.17%	76.23
H&M11	86.67%	74.71%	80.25
TOTAL	83.49%	79.13%	81.25

Table 3: Geoparsing: results per article; 575 phrases; found: 545; correct: 455. Articles: L09 (Linke 2009), H09 (Hammerstaedt 2009), S10 (Sanger 2010), H&M11 (Haensch and Mackensen 2011)

The performances of the geoparser, on the other hand, seem encouraging (Table 3). With gold-standar named entity recognition, the Edinburgh Geoparsers combined with the `idai.gazetteer` attained scores that closely approximate, or even surpass 80%. The evaluation of our annotation was also a valuable occasion to assess the accuracy and granularity of the `idai.gazetteer`: 38 locations in North Africa mentioned in one article (Haensch and Mackensen, 2011) did not have any record in DAI's gazetteer.

3.3 Applying in-domain NER models

We decided to use the manually corrected articles to see whether we could improve on the baseline with the help of in-domain models. We trained a CRF model adding a series of linguistic features, like POS, which may help capturing non-German expressions, or type-set features such as the use of small- and full-caps.[12] As the articles in *Chiron* focus on the Greco-Roman civilization, we expect a lookup in lists of known toponyms of the Ancient Word to sensibly improve the performances of NER for locations. We chose to add a gazetteer lookup to the list of features; we preferred to resort to a more specific resource like the "Digital Atlas of the Roman Empire" (DARE)[13] instead of the general-purpose `idai.gazetteer`.

Entity	Precision	Recall	$F_{\beta=1}$
Person	80.00%	71.41%	75.30
Location	76.26%	58.90%	65.87
Organization	22.02%	23.08%	16.94
TOTAL	79.32%	65.75%	71.75

Table 4: NER: results of the in-domain model; average scores of 10-fold cross-validation

Table 4 reports the results of this second round of testing, which was conducted using the same methodology as before and performing a 10-fold cross-validation. As can be seen, the in-domain model considerably improves over the baseline. The performance with organizations is still largely insufficient, mainly on account of the scarcity of examples (70 phrases, vs 970 persons, 387 locations). The improvement with locations is significant, but the overall performance still leaves room for substantial improvement.

4 Conclusions and future work

The use of in-domain CRF models trained specifically for the target journal and adopting a specialized gazetteer for place names improves on the baseline of the out-of-the-box NER tools in our initial pipeline. It is likely that the accuracy on the *Chiron* data can be further increased with additional training. Given that an accurate recognition is a prerequisite for geoparsing, we plan to con-

[12] The CRF implementation that we used is provided by the Python library `sklearn-crfsuite` (0.3.6).

[13] `http://dare.ht.lu.se/`

centrate our effort on the NER components. We intend to progress in the direction discussed above, in particular by: a. training and evaluating models for the other languages (French, English, Italian, Spanish) b. testing the models on other publications in the portal.

In a more distant future, we also intend to include support to the identification (and subsequent linking) of other named entities of interest for archaeologists, such as artifacts, monuments and chronological references.

References

Beatrice Alex. 2017. Geoparsing English-Language Text with the Edinburgh Geoparser. https://programminghistorian.org/en/lessons/geoparsing-text-with-edinburgh.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly, New York.

Manaal Faruqui and Sebastian Padó. 2010. Training and evaluating a german named entity recognizer with semantic generalization. In *Proceedings of KONVENS 2010*, Saarbrücken, Germany.

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 on Association for Computational Linguistics*, ACL '05, pages 363–370, Stroudsburg, PA, USA. Association for Computational Linguistics.

Claire Grover, Richard Tobin, Kate Byrne, Matthew Woollard, James Reid, Stuart Dunn, and Julian Ball. 2010. Use of the Edinburgh geoparser for georeferencing digitized historical collections. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 368:3875–3889.

Rudolf Haensch and Michael Mackensen. 2011. Das tripolitanische Kastell Gheriat el-Garbia im Licht einer neuen spätantiken Inschrift: Am Tag, als der Regen kam. *Chiron*, 41:263–286.

Jürgen Hammerstaedt. 2009. Warum Simonides den Artemidorpapyrus nicht hätte fälschen können: Eine seltene Schreibung für Tausender in Inschriften und Papyri. *Chiron*, 39:323–338.

Bernhard Linke. 2009. Jupiter und die Republik. Die Entstehung des europäischen Republikanismus in der Antike. *Chiron*, 39:339–358.

Marco Lui and Timothy Baldwin. 2011. Cross-domain feature selection for language identification. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 553–561, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

Johannes Nollé and Sylvia Wartner. 1987. Ein tückischer Iotazismus in einer milesischen Inschrift. *Chiron*, 17:361–364.

A. Palmero Aprosio and G. Moretti. 2016. Italy goes to Stanford: a collection of CoreNLP modules for Italian. *ArXiv e-prints*.

Patrick Sänger. 2010. Kommunikation zwischen Prätorianerpräfekt und Statthalter: Eine Zweitschrift von IvE Ia 44. *Chrion*, 40:89–102.

Helmut Schmid. 1999. Improvements in Part-of-Speech Tagging with an Application to German. In Susan Armstrong, Kenneth Church, Pierre Isabelle, Sandra Manzi, Evelyne Tzoukermann, and David Yarowsky, editors, *Natural Language Processing Using Very Large Corpora*, volume 11 of *Text, Speech and Language Processing*, pages 13–26. Kluwer Academic Publishers, Dordrecht.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 Shared Task: Chunking. In *Proceedings of the 2Nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Learning - Volume 7*, ConLL '00, pages 127–132, Stroudsburg, PA. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of CoNLL-2003*, pages 142–147. Edmonton, Canada.

Source-driven Representations for Hate Speech Detection

Flavio Merenda[*∓], **Claudia Zaghi**[*], **Tommaso Caselli**[*], **Malvina Nissim**[*]
[*] Rikjuniversiteit Groningen, Groningen, The Netherlands
[∓] Università degli Studi di Salerno, Salerno, Italy
f.merenda|t.caselli|m.nissim@rug.nl c.zaghi@student.rug.nl

Abstract

English. Sources, in the form of selected Facebook pages, can be used as indicators of hate-rich content. Polarized distributed representations created over such content prove superior to generic embeddings in the task of hate speech detection. The same content seems to carry a too weak signal to proxy silver labels in a distant supervised setting. However, this signal is stronger than gold labels which come from a different distribution, leading to re-think the process of annotation in the context of highly subjective judgments.

Italiano. *La provenienza di ciò che viene condiviso su Facebook costituisce un primo elemento indentificativo di contentuti carichi di odio. La rappresentazione distribuita polarizzata che costruiamo su tali contenuti si dimostra migliore nell'individuazione di argomenti di odio rispetto ad alternative più generiche. Il potere predittivo di tali embedding polarizzati risulta anche più incisivo rispetto a quello di dati gold standard che sono caratterizzati da una distribuzione ed una annotatione diverse.*

1 Introduction

Hate speech is "the use of aggressive, hatred or offensive language, targeting a specific group of people sharing a common trait: their gender, ethnic group, race, religion, sexual orientation, or disability" (Merriam-Webster's collegiate dictionary, 1999). The phenomenon is widely spread on-line, and Italian Social Media is definitely not an exception (Gagliardone et al., 2015). To monitor the problem, social networks and websites have introduced a stricter code of conduct and regularly remove hateful content flagged by users (Bleich, 2014). However, the volume of data requires that ways are found to classify on-line content automatically (Nobata et al., 2016; Kennedy et al., 2017).

The Italian NLP community is active on this front (Poletto et al., 2017; Del Vigna et al., 2017), with the development of labeled data, including the organization of a dedicated shared task at the EVALITA 2018 campaign[1]. Relying on manually labeled data has limitations, though: i.) annotation is time and resource consuming; ii.) portability to new domains is scarce[2]; iii.) biases are unavoidable in annotated data, especially in the form of annotation decisions. This is both due to the intrinsic subjectivity of the task itself, and to the fact that there is not, as yet, a shared set of definitions and guidelines across the different projects that yield annotated datasets.

Introduced as a new take on data annotation (Mintz et al., 2009; Go et al., 2009), *distant supervision* is used to automatically assign (silver) labels based on the presence or absence of specific hints, such as happy/sad emoticons (Go et al., 2009) to proxy positive/negative labels for sentiment analysis, Facebook reactions (Pool and Nissim, 2016; Basile et al., 2017) for emotion detection, or specific strings to assign gender (Emmery et al., 2017). Such an approach has the advantage of being more scalable (portability to different languages or domains) and versatile (time and resources needed to train), than pure supervised learning algorithms, while preserving competitive performance. Apart from the ease of generating labeled data, distant supervision has a valuable *ecological* aspect in not relying on third-party annotators to interpret the data (Purver and Battersby,

[1] http://www.di.unito.it/~tutreeb/haspeede-evalita18/index.html

[2] The EVALITA 2018 haspeede task addresses this issue by setting the task in a cross-genre fashion.

2012). This reduces the risk of adding extra bias (see also point (iii) about limitation in the previous paragraph), modulo the choices related to which proxies should be considered.

Novelty and Contribution We promote a special take on distant supervision where we use as proxies the *sources where the content is published on-line rather than any hint in the content itself*. Through a battery of experiments on hate speech detection in Italian we show that this approach yields meaningful representations and an increase in performance over the use of generic representations. Contextually, we show the limitations of silver labels, but also of gold labels that come from a different dataset with respect to the evaluation set.

2 Source-driven Representations

Our approach is based on previous studies on on-line communities showing that communities tend to reinforce themselves, enhancing "filter bubbles" effects, decreasing diversity, distorting information, and polarizing socio-political opinions (Pariser, 2011; Bozdag and van den Hoven, 2015; Seargeant and Tagg, 2018). Each community in the social media sphere thus represents a somewhat different source of data. Our hypothesis is that the contents generated by each community (source) can thus be used as proxies for specialized information or even labeled data.

Building on this principle, we scraped data from social media communities on Facebook, acquiring what we call *source-driven representations*. The data is indeed used in two ways in the context of Hate Speech detection, namely: i.) to generate (potentially) *polarized word embeddings* to be used in a variety of models, comparing it to more standard generic embeddings (Section 3); and ii.) as *training data* for a supervised machine learning classifier, combining and comparing it with manually labeled data (Section 4).

3 Polarized Embeddings

Polarized embeddings are representations built on a corpus which is not randomly representative of the Italian language, rather collected with a specific bias. In this context, we use data scraped from Facebook pages (communities) in order to create hate-rich embeddings.

Data acquisition We selected a set of publicly available Facebook pages that may promote or be the target of hate speech, such as pages known for promoting nationalism (*Italia Patria Mia*), controversies (*Dagospia*, *La Zanzara - Radio 24*), hate against migrants and other minorities (*La Fabbrica Del Degrado*, *Il Redpillatore*, *Cloroformio*), support for women and LGBT rights (*NON UNA DI MENO*, *LGBT News Italia*). Using the Facebook API, we downloaded the comments to posts as they are the text portions most likely to express hate, collecting a total of over 1M comments for almost 13M tokens (Table 1).

Page Name	Comments
Matteo Salvini	318,585
NON UNA DI MENO	5,081
LGBT News Italia	10,296
Italia Patria Mia	4,495
Dagospia	41,382
La Fabbrica Del Degrado	6,437
Boom. Friendzoned.	85,132
Cloroformio	392,828
Il Redpillatore	6,291
Sesso Droga e Pastorizia	8,576
PSDM	44,242
Cara, sei femminista - Returned	830
Se solo avrei studiato	38,001
La Zanzara - Radio 24	215,402
Total	**1,177,578**

Table 1: List of public pages from Facebook and number of extracted comments per page.

Making Embeddings We built distributed representations over the acquired data. The embeddings have been generated with the `word2vec` [3] skip-gram model (Mikolov et al., 2013) using 300 dimensions, a context window of 5, and minimum frequency 1. The final vocabulary amounts to 381,697 words.

These hate-rich embeddings are used in models for hate speech detection. For comparison, we also use larger, generic embeddings that were trained on the Italian Wikipedia (more than 300M tokens)[4] using `GloVe` (Berardi et al., 2015)[5]; the vocabulary amounts to 730,613 words. As a sanity check, and a sort of qualitative intrinsic evaluation, we probed our embeddings with a few keywords, reporting in Table 2 the top three nearest neighbors for the words "immigrati" [migrants]

[3] `https://radimrehurek.com/gensim/;https://github.com/RaRe-Technologies/gensim`

[4] `http://hlt.isti.cnr.it/wordembeddings/`

[5] `https://nlp.stanford.edu/projects/glove/`

and "trans". For the former, it is interesting to see how the polarized embeddings return more hate-leaning words compared to the generic embeddings. For the latter, in addition to hateful epithets, we also see how these embeddings capture the correct semantic field, while the generic ones do not.

Table 2: Intrinsic embedding comparison: words most similar to potential hate targets.

Generic Embeddings	Polarized Embeddings
"immigrati" [migrants]	
immigranti (0.737)	extracomunitari (0.841)
emigranti (0.731)	immigranti(0.828)
emigrati (0.725)	clandestini (0.823)
"trans" [trans]	
europ (0.399)	lesbo (0.720)
express (0.352)	puttane (0.709)
airlines (0.327)	gay (0.703)

Classification To test the contribution of our embeddings, we used them in two different classifiers, comparing them to alternative distributed representations.

First, we built a Convolutional Neural Network (CNN), using the implementation of (Kim, 2014). This is a simple architecture with one convolutional layer built on top of a word embeddings layer (hyperparameters: `Number of filters`: 6; `Filter sizes`: 3, 5, 8; `Strides`: 1; `Activation function`: Rectifier). We experimented with three different activation strategies for the CNN model: i.) random initialization, by generating word embeddings from the training data itself, i.e. "on-the-fly"; ii.) pre-trained 300 dimension general word embeddings; iii.) our own polarised embeddings.

Second, and for further comparison, we also built a simple Linear Support Vector Machine (SVM), using the LinearSVC scikit learn implementation (Pedregosa et al., 2011). In one setting, we used only information coming from the two different sets of pre-trained embeddings (GloVe generic vs our polarized ones) to observe their contribution alone, in the same fashion as the CNN. To use these word vectors in the SVM model, we mapped the content words in each sentence and we replaced them with the corresponding word embeddings values; afterwards, we computed the average value for each word embedding, in order to achieve a unique one-dimensional sentence vector with each word replaced with the corresponding embedding average. In further settings, we combined this information with a more standard n-gram-based tf-idf model. Specifically, we use 1-3 word and 2-4 character n-grams, with default parameter values for the SVM.

We train and test our models using the manually labelled data provided in the context of the EVALITA 2018 task on Hate Speech Detection (`haspeede`) [6]. The released training/development set comprises 3000 Facebook comments and 3000 tweets. The proportion of hateful content in this dataset is 39%, with 46% in the Facebook portion, and 32% in Twitter. We train on 80% of `haspeede` (4800 instances), and test on the remaining 20%. We report precision, recall, and F-score per class, averaged over ten random train/test splits. To assess general performance, we use macro F-score rather than micro F-score as the classifier's accuracy on the minority class is particularly important. This is also reported as the average of the ten different runs.

Results The results in Table 3 show that despite our embeddings being almost 25 times smaller than the generic ones, they yield a substantially better performance both in the CNN model and in the SVM classifier. In the former, they are also more informative than the representations obtained on-the-fly from the training data. In the latter, the contribution of embeddings in general appears though rather marginal on top of a more standard SVM model based on n-gram tf-idf information, and the difference according to which representation is used is not significant. Finally, it is interesting to note that the polarized embeddings cover 55% of the tokens in the training data (*vs*. only 45% of the generic ones, in spite of the substantial size difference between the two.

4 Silver labels

In a more standard distantly supervised setting, modulo proxing labels via sources rather than specific keywords/emojis, we also used the scraped text as training data directly. Because we approximate labels with sources, and we had collected data from supposedly hate-rich pages, for the current experimental settings we balanced the data by

[6] `http://www.di.unito.it/~tutreeb/haspeede-evalita18/index.html`

Table 3: Results for the contribution of different embeddings in CNN and SVM models. The models are trained and tested on 80/20 splits randomised ten times on manually labelled data. Results are reported as averages. We underline the best score for each set of experiments, and boldface the best score overall.

MODEL	CLASS	P	R	F	MACRO F
EMBEDDINGS ALONE					
CNN on-the-fly embeds	non-H	.84	.75	.79	.749
	H	.77	.65	.70	
CNN generic embeds	non-H	.80	.86	.83	.760
	H	.74	.65	.69	
CNN polarised embeds	non-H	.82	.88	.85	<u>.786</u>
	H	.78	.68	.73	
SVM generic embeds	non-H	.77	.85	.81	.728
	H	.71	.60	.65	
SVM polarised embeds	non-H	.79	.84	.81	<u>.750</u>
	H	.72	.66	.69	
N-GRAMS + EMBEDDINGS					
SVM tf-idf + generic embeds	non-H	.84	.87	.85	.806
	H	.78	.74	.76	
SVM tf-idf + polarised embeds	non-H	.84	.86	.85	**<u>.807</u>**
	H	.78	.75	.76	
N-GRAMS ALONE					
SVM tf-idf	non-H	.83	.87	.85	.802
	H	.78	.72	.75	

scraping Facebook comments from an Italian news agency (i.e. ANSA), assuming it conveys neutral content rather than polarized.

As for the distribution of labels, we followed the proportion of the Facebook portion of the haspeede dataset (46% of hateful content, and the rest non-polarized). We proxy labels according to sources, and under the above presumed proportions, we selected a total of 100,000 comments.

For comparison, and in combination, we also used gold data. In addition to the previously mentioned 6000 instances from the haspeede task, we used the Turin dataset, a collection of 990 manually labelled tweets concerning the topic of immigration, religion and Roma[7] (Poletto et al., 2017; Poletto et al., 2018). The distribution of labels in this dataset differs from the EVALITA dataset, with only 160 (16%) hateful instances.

We trained an SVM classifier with the best settings as observed in Section 3 (tf-idf and and polarised embeddings) using different training sets, combining gold and silver data (see Table 4). For evaluation, we use the same settings as the experiments in Section 3, by picking a random test set out of the haspeede dataset ten times, and reporting averaged results.

Table 4: Evaluation on 1200 instances from haspeede (averaged over 10 randomly picked test sets), using train sets from different sources and combinations thereof. The haspeede and Turin sets have gold labels.

TRAINSET	CLASS	P	R	F	MACRO F
100K silver	non-H	.60	.39	.47	.464
	H	.38	.59	.46	
3600 haspeede	non-H	.85	.86	.85	.807
	H	.77	.76	.76	
3600 haspeede + 1000 silver	non-H	.83	.85	.84	.792
	H	.76	.73	.74	
3600 haspeede + 990 Turin	non-H	.81	.86	.83	.777
	H	.76	.68	.72	
3600 haspeede + 1200 haspeede	non-H	.85	.86	.85	.814
	H	.78	.77	.77	

Results From Table 4 we can make the following observations: (i) training on silver labels lets us detect hate speech better than a most-frequent-label baseline (macro F=.383); (ii) however, in this context, training on small amounts of gold data is substantially more accurate than training on large amounts of distantly supervised data (.807 vs .464); (iii) adding even small amounts of silver data to gold decreases performance (.792 vs .807)[8]; (iv) also adding more gold data decreases performance, *even more so than adding an equal amount of silver data*, if the manually labeled data comes from a different dataset (thus created with different guidelines, and in this case with a different hate/non-hate distribution). Performance goes up as expected when adding more data from the same dataset (.814 vs .807).

5 Conclusions

We exploited distant supervision to automatically obtain representations from Facebook-scraped content in two forms. First, we generated polarized, hate-rich distributed representations which proved superior to larger, generic embeddings when used both in a CNN and an SVM model for hate speech detection. Second, we used the scraped data as training material directly, proxing

[7]The Romani, Romany, or Roma are an ethnic group of traditionally itinerant people who originated in northern India and are nowadays subject to ethnic discrimination.

[8]We also experimented with adding progressively larger batches of silver data to gold (2K, 3K, 5K, etc.), but this yielded a steady decrease in performance.

labels (hate vs non-hate) with the sources where the data was coming from (Facebook pages). This did not prove as a successful alternative nor complementary strategy to using gold data, though performance above baseline indicates some signal is present. Importantly, though, our experiments also suggest that gold data is not better than silver data if it comes from a different dataset. This highlights a crucial aspect related to the creation of manually labeled datasets, especially in the highly subjective area of hate speech and affective computing in general, where different guidelines and different annotators clearly introduce large biases and discrepancies across datasets.

All considered, we believe that obtaining data in a distant, more ecological way should be further pursued and refined. How to better exploit the information that comes from polarized embeddings in combination with other features is also left to future work.

Acknowledgments

The authors want to thank the EVALITA 2018 Hate Speech Detection (HaSpeeDe) task organizers for allowing us to use their datasets.

References

Angelo Basile, Tommaso Caselli, and Malvina Nissim. 2017. Predicting Controversial News Using Facebook Reactions. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017)*, Rome, Italy.

Giacomo Berardi, Andrea Esuli, and Diego Marcheggiani. 2015. Word embeddings go to italy: A comparison of models and training datasets. In *IIR*.

Erik Bleich. 2014. Freedom of expression versus racist hate speech: Explaining differences between high court regulations in the usa and europe. *Journal of Ethnic and Migration Studies*, 40(2):283–300.

Engin Bozdag and Jeroen van den Hoven. 2015. Breaking the filter bubble: democracy and design. *Ethics and Information Technology*, 17(4):249–265.

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), Venice, Italy, January 17-20, 2017*, pages 86–95.

Chris Emmery, Grzegorz Chrupała, and Walter Daelemans. 2017. Simple queries as distant labels for predicting gender on twitter. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 50–55.

Iginio Gagliardone, Danit Gal, Thiago Alves, and Gabriela Martinez. 2015. *Countering online hate speech*. Unesco Publishing.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1(12).

George Kennedy, Andrew McCollough, Edward Dixon, Alexei Bastidas, John Ryan, Chris Loo, and Saurav Sahay. 2017. Technology solutions to combat online harassment. In *Proceedings of the First Workshop on Abusive Language Online*, pages 73–77.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. 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: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, pages 145–153. International World Wide Web Conferences Steering Committee.

Eli Pariser. 2011. *The filter bubble: What the Internet is hiding from you*. Penguin UK.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Fabio Poletto, Marco Stranisci, Manuela Sanguinetti, Viviana Patti, and Cristina Bosco. 2017. Hate speech annotation: Analysis of an italian twitter corpus. In *CEUR WORKSHOP PROCEEDINGS*, volume 2006, pages 1–6. CEUR-WS.

Fabio Poletto, Cristina Bosco, Viviana Patti, and Marco Stranisci. 2018. An italian twitter corpus of hate speech against immigrants. *In Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*.

Chris Pool and Malvina Nissim. 2016. Distant supervision for emotion detection using facebook reactions. In *Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES)*, pages 30–39, Osaka, Japan, December. COLING 2016.

Matthew Purver and Stuart Battersby. 2012. Experimenting with distant supervision for emotion classification. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 482–491. Association for Computational Linguistics.

Philip Seargeant and Caroline Tagg. 2018. Social media and the future of open debate: A user-oriented approach to Facebook's filter bubble conundrum. *Discourse, Context & Media*.

Progettare chatbot: considerazioni e linee guida

Eleonora Mollo
Università degli Studi di Torino
325545@edu.unito.it

Amon Rapp
Università degli Studi di Torino
amon.rapp@gmail.com

Dario Mana
TIM
dario.mana@telecomitalia.it

Rossana Simeoni
TIM
rossana.simeoni@telecomitalia.it

Abstract

Italiano. Il lavoro si propone di delineare una serie di linee guida per la progettazione di chatbot e assistenti virtuali a partire dall'analisi degli attuali trend di progettazione e delle esigenze lato utente rilevate da precedenti lavori di rassegna della letteratura esistente. Il presente lavoro è stato svolto nell'ambito del progetto "Cognitive Solution for Intelligent Caring" di TIM.

English. *This work is focused on the current trends in designing chatbots and virtual assistants. We start from users' needs identified in industrial surveys on chatbots. The result is a collection of guidelines and considerations which reflect the state of the art.*

1 Introduzione

Chatbot e assistenti virtuali sono un ambito in via di sviluppo. Numerose aziende si stanno muovendo per sincronizzare le proprie funzioni di marketing, vendite e assistenza in modo da offrire ai propri utenti un'esperienza positiva che incontri le loro aspettative durante l'interazione. Secondo una ricerca condotta da Oracle, "Can virtual Experience replace reality"(Oracle, 2016), brand B2B e B2C hanno compreso che ci sono ampi margini per migliorare le proprie attività grazie al supporto dell'intelligenza artificiale: tra le loro priorità c'è sicuramente un potenziamento della Customer Experience (CX). Il 78% dei brand intervistati hanno implementato o hanno programmato di indirizzare entro il 2020 investimenti in Intelligenza Artificiale (IA) o in Realtà Virtuale. Proprio in ottica di una migliore CX, la presente ricerca ha l'obiettivo di analizzare le attuali strategie per la costruzione di chatbot e assistenti virtuali. Vengono di seguito delineati i bisogni degli utenti in merito all'interazione con i chatbot tramite una rassegna di survey condotte da importanti player industriali, quali Capgemini (Capgemini, 2017) e Amdocs (Amdocs, 2017). Successivamente si è svolta un'indagine per capire quali siano le caratteristiche che gli assistenti virtuali dovrebbero possedere, delineando trend emergenti riguardanti "buone pratiche" di progettazione. Il risultato è una serie di indicazioni che un progettista dovrebbe perseguire nel momento in cui si propone di costruire un chatbot capace di soddisfare la CX. Con l'obiettivo di un'esplorazione preliminare del campo e non di una review sistematica, la ricerca è stata condotta utilizzando Google Scholar a inizio 2018 con le seguenti parole chiave: *conversational interface*, *invisible ui*, *no ui*, *assistente digitale*, *chatbot*, *chatops*, *scrollytelling*, *design patterns conversational*, *question answering*.

2 Esigenze degli utenti tratte da survey condotte da player industriali

L'indagine parte dall'analisi di tre survey realizzate da i) Capgemini (Capgemini, 2017), società attiva nel settore della consulenza in ambito informatico, ii) Amdocs (Amdocs, 2017), provider di servizi informatici per attività di comunicazione e media, e iii) Chatbot.org (Chatbot.org, 2018), sito web specializzato in assistenti virtuali, realizzate per analizzare l'approccio utente-chatbot e per capire il loro possibile utilizzo futuro. Sono state scelte queste survey in quanto mettono in luce le reali esigenze che gli utenti si trovano ad affrontare quando interagiscono con questo tipo di strumenti. Lo studio condotto da Capgemini (Capgemini, 2017) mette in luce il grande progresso che gli assistenti virtuali hanno

avuto nel tempo; sono gli stessi utenti a confermare il trend in atto: entro i prossimi tre anni (la ricerca è datata Ottobre-Novembre 2017) si suppone che il numero di consumatori che preferirà rivolgersi ad un assistente piuttosto che andare in un negozio fisico raddoppierà. Un altro dato interessante, e che dovrà essere tenuto in considerazione nel momento in cui si intende realizzare un chatbot, riguarda il tipo di device che viene utilizzato: lo smartphone risulta essere il mezzo più comune, pertanto sarà necessario valutare l'eventuale utilizzo di spazi sullo schermo, virtual keyboard ecc. In generale gli utenti sono soddisfatti di come avvengono le transazioni attraverso chatbot. Le caratteristiche più apprezzate, e che vengono maggiormente ricercate nel momento in cui si decide di utilizzare un assistente virtuale, sono la velocità, l'automatizzazione della routine d'acquisto, la personalizzazione, il risparmio di tempo e di soldi. Ancora oggi, però, permane l'esigenza da parte dei consumatori di interfacciarsi con un agente umano, in quanto si pensa che possa comprendere meglio le loro esigenze e sia maggiormente empatico rispetto all'umore dell'utente. Una delle sfide che i progettisti di interfacce conversazionali dovranno affrontare riguarda proprio questo aspetto: i soggetti si aspettano che le interazioni con i chatbot si avvicinino quanto più possibile a quelle con gli umani, quindi si riconoscono esigenze di "umanità" (senza cadere nel fenomeno dell'uncanny valley (Ciechanowski et al., 2018)), empatia, buone maniere. Tra i problemi principali si trovano la necessità di sicurezza e di protezione dei dati personali; inoltre, c'è poca fiducia nel fatto che gli assistenti sappiano correttamente interpretare le esigenze degli individui.

Lo studio di Amdocs (Amdocs, 2017) contribuisce a dettagliare ulteriormente le esigenze espresse dai consumatori. Gli utenti vogliono essere al centro del brand, quindi avere interazioni sempre più personalizzate e modellate sui loro bisogni, devono poter utilizzare diversi canali per mettersi in contatto con l'azienda, soprattutto utilizzando lo smartphone. Anche in questo caso i motivi principali per cui si decide di operare tramite assistenti virtuali vanno ricercati nella velocità e nell'automazione. Problemi, invece, sono stati riscontrati nell'incapacità dei chatbot nel risolvere questioni complesse oppure nella mancanza di empatia durante l'interazione.

Per quanto riguarda la ricerca condotta da Chatbot.org (Chatbot.org, 2018), il dato più rilevante riguarda la frustrazione che gli utenti incontrano nel momento in cui la conversazione passa da un assistente virtuale a un agente umano: è spesso fonte di stress il fatto di dover ripetere una serie di informazioni precedentemente comunicate al chatbot.

3 Analisi dei trend

Nella analisi di quali siano le indicazioni per la costruzione di assistenti virtuali è stato possibile riscontrare una varietà di linee guida. Ciascuno di questi trend trova una formalizzazione più o meno forte all'interno della letteratura di tipo accademico, rintracciata tra i contributi più recenti nell'ambito della costruzione di assistenti virtuali. Altre indicazioni, invece, derivano da applicazioni di tipo pratico come suggerimento per la buona progettazione.

Ogni chatbot presenta delle caratteristiche generali che devono essere sempre realizzate:

3.1 Soluzione invisibile ai problemi

Gli utenti non vogliono sapere come un assistente virtuale raggiungerà la soluzione. Per le necessità di velocità e semplicità, il consumatore non deve sapere quali siano i meccanismi che sottendono alla soluzione del bisogno presentato (Accenture Interactive, 2017; Fadhil, 2018).

3.2 Conoscenza dei path

Per ottenere una risposta nel più breve tempo possibile è importante che i chatbot abbiano ben chiaro quale sia il percorso da seguire per raggiungere la soluzione. Avere un disegno netto, lineare delle opzioni più rilevanti per ciascuna richiesta proveniente dall'utente è fondamentale (Fadhil, 2018; Daniel et al., 2018).

3.3 Focus su questioni specifiche

Il chatbot migliore è quello che si concentra su un argomento in particolare. Spesso ci troviamo di fronte ad assistenti onniscienti, ma quelli che si muovono attorno a un ambito piuttosto ristretto di tematiche hanno prestazioni migliori perché il range di questioni che vengono poste di volta in volta è limitato a pochi argomenti (Accenture Interactive, 2017; Action on Google, 2018; Fadhil, 2018).

3.4 Capacità di predizione

Questo punto è strettamente collegato con la necessità di personalizzazione che gli utenti richiedono agli assistenti virtuali. Se i chatbot conoscono, grazie ad interazioni precedenti oppure alle informazioni che possono acquisire da un

database, con chi stanno parlando, essi potrebbero addirittura predire le scelte che si effettueranno. Si tratta quindi di conoscere le preferenze e saper anticipare ciò che i propri consumatori desiderano (Fadhil, 2018; Daniel et al., 2018).

3.5 Riduzione del carico cognitivo

In questo caso si valuta l'importanza nell'utilizzare correttamente UI components come immagini, bottoni, carousel, quick reply. Questi escamotage possono essere utilizzati per indirizzare la conversazione e rendere più agevole sia per l'utente che per il bot la costruzione di un'interazione che sia soddisfacente per l'uno e gestibile per l'altro (Fadhil, 2018; Valério et al., 2017; Knutsen et al., 2016; Kevin, 2016).

3.6 Comprensione del contesto d'uso e dispositivo

Nel momento in cui si intende progettare un bot bisognerebbe fare un'analisi di quali siano i dispositivi su cui vengono utilizzati e i contesti di maggior impiego. A seconda del luogo in cui il chatbot verrà usato si dovranno implementare determinate funzionalità e caratteristiche. Secondo le ricerche di Capgemini (Capgemini, 2017) e Amdocs (Amdocs, 2017) il dispositivo più utilizzato è lo smartphone, pertanto si dovranno tenere in considerazione limitazioni di spazio dello schermo dato, dato che la tastiera da sola ne occupa la metà. Per questo motivo è necessario evitare di scrivere testi lunghi per scongiurare il rischio di scrolling. Quindi meglio suddividere la conversazione in brevi, ma efficaci, interazioni, oppure reindirizzare l'utente verso un sito terzo (Begany et al., 2015; Bianchini, 2017; Daniel et al., 2018; Morrissey et al., 2013; Oracle, 2016).

3.7 Antropomorfizzazione

Le conversazioni devono essere human-like, quindi rispettare i canoni della comunicazione tra esseri umani. Gli utenti apprezzano interagire con bot che abbiano tratti riconducibili a quelli umani, ma senza arrivare all'eccessivo realismo. Il pericolo che si corre è quello di cadere nel fenomeno dell'uncanny valley, ovvero un'antropomorfizzazione eccessiva che muove nel soggetto addirittura dei sentimenti di disgusto e repulsione. Per evitare questo fenomeno i chatbot possono essere rappresentati in chiave fumettistica, giocando con rappresentazioni grafiche (Araujo, 2018; Ciechanowski et al., 2018; Kangsoo et al., 2018; Luger et al., 2016; Vinayak et Arpit, 2018; Eunji, 2017).

3.8 Sicurezza

Il tema della sicurezza è sicuramente uno dei più importanti per gli utenti, in quanto affidano i propri dati sensibili a degli agenti digitali che non possono controllare. È importante, quindi, che i chatbot risultino affidabili e che non scherzino con un patrimonio così prezioso (Eunji, 2017; Limerick et al., 2015; Luger et Sellen, 2016; Microsoft, 2018; Van Eeuwen, 2018).

3.9 Prima interazione

Il primo approccio con un assistente virtuale può condizionare l'andamento di tutta la conversazione e rappresenta, quindi, un passaggio fondamentale. La prima interazione può essere messa in atto facendo in modo che il bot si presenti e metta subito in mostra le proprie funzionalità; utilizzando bottoni/menu/carousel che presentano le azioni realizzabili. Iniziare con affermazioni troppo generiche non aiuta; partire, invece, con un menu può essere un buon preludio all'interazione (Microsoft, 2018; Valério et al., 2017).

Ulteriori indicazioni utili per la realizzazione di chatbot riguardano più nello specifico il design della conversazione tra uomo e macchina:

3.10 Comprensione del linguaggio naturale

Caratteristica necessaria è ovviamente la comprensione del linguaggio naturale. Capacità tutt'altro che scontata dato che spesso le espressioni umane sono denotate da slang, dialetti, frasi fatte, complicando la comprensione dell'utente. In questo frangente vediamo che suggerimenti provenienti da menu, carousel, quick replies possono venire in aiuto nel rendere l'interazione più agevole (Daniel et al., 2018; Fadhil, 2018; Fourault, 2017; Microsoft, 2018).

3.11 Input validation/feedback

Gli input inviati al chatbot devono venire in qualche maniera validati da parte di quest'ultimo. È possibile chiedere una conferma all'utente, o ripetere le informazioni che sono state inserite (specialmente se riguardano dei pagamenti). Grazie a questo meccanismo si riesce a conferire un grado di maggior sicurezza alle persone, infondendo maggior fiducia nelle potenzialità del bot. Al termine della conversazione, inoltre, può essere utile richiedere all'utente se sia soddisfatto dell'interazione oppure se abbia dei consigli per migliorarla (Action on Google, 2018; Begany et al., 2015; Fadhil, 2018; Luger et Sellen, 2016).

3.12 Utilizzo dei menu a bottoni

La funzione di menu a bottoni e quick replies è già stata esplicitata, in quanto essi rappresentano una possibile chiave di una navigazione semplice ed efficace. Da notare che sussiste una differenza tra di essi: i bottoni non spariscono nel procedere della conversazione, mentre le quick replies sì. Nell'economia dell'interazione andrebbe valutato attentamente quale di questi componenti utilizzare: se dare la possibilità all'utente di tornare indietro e cambiare le proprie preferenze oppure effettuare una nuova domanda (Eunji, 2017; Fadhil, 2018; Fourault, 2017; Microsoft, 2018; Mohit et al., 2018).

3.13 Conversazioni lineari e corte

Il discorso dovrebbe procedere con linearità senza incappare in divagazioni, quindi non aprire nuovi argomenti, ma procedere a senso unico con un botta e risposta tra utente e bot. Ovviamente le conversazioni devono essere le più concentrate possibili, focalizzandosi su un dominio particolare di problemi e risolvendo in modo puntuale le questioni proposte (Action on Google, 2018; Eunji, 2017; Fadhil, 2018).

3.14 Turn taking

Per ottenere un effetto human-like è opportuno che la conversazione si svolga in modalità di botta e risposta. Evitare, quindi, di far dare al bot una serie di risposte in sequenza senza permettere all'utente di replicare (Action on Google, 2018).

3.15 Conoscenza del contesto linguistico

Questo è un tratto particolarmente problematico, soprattutto in contesti fortemente caratterizzati da varietà linguistica e dialettale. Il bot deve poter essere in grado di interpretare correttamente richieste che spesso non vengono formulate in italiano corretto (Action on Google, 2018; Eunji, 2017; Kevin, 2016; Mohit et al., 2018).

3.16 Flessibilità

Il bot deve avere a disposizione un'ampia varietà di risposte in modo da non risultare pedante nelle proprie affermazioni (Action on Google, 2018; Daniel et al., 2018; Eunji, 2017; Fadhil, 2018; Kevin, 2016).

3.17 Gestire gli errori e fornire una way out

Per non mandare in confusione l'utente e per garantire una certa fiducia nell'assistente virtuale una corretta gestione degli errori è importante. Ogni volta che l'utente commette un "errore", il bot deve rispondere in modo preciso, variando nelle proprie risposte e offrendo sempre una scappatoia. L'individuo deve anche essere messo nelle condizioni di tornare indietro qualora lo ritenga necessario (Action on Google, 2018; Fadhil, 2018; Eunji, 2017; Kevin, 2016).

3.18 Precedenti conversazioni visibili

Per garantire anche una personalizzazione della conversazione, può risultare utile tenere traccia delle interazioni precedenti, in modo che l'utente possa recuperare le informazioni in caso di necessità (Daniel et al., 2018; Mohit et al., 2018).

3.19 Chiudere le conversazioni in modo opportuno

Al termine della conversazione l'utente deve essere invogliato a fare nuovamente uso del bot, quindi il suo uso deve interrompersi in modo piacevole e magari invitare ad utilizzare altre funzionalità (Action on Google, 2018; Eunji, 2017).

3.20 Gestione dell'attesa

Rispetto ad altre applicazioni, l'utente quando interagisce con un assistente virtuale è disposto ad aspettare fino ad 8 secondi prima di ottenere una risposta. Nel caso l'attesa si protraesse nel tempo, è anche possibile utilizzare degli espedienti grafici come i typing indicator per mostrare che il bot è ancora attivo e sta lavorando (Eunji, 2017).

Infine, vengono valutate le caratteristiche legate alla personalità del bot che contribuiscono a rendere empatica e naturale la conversazione:

3.21 Buone maniere e presentazioni

Il chatbot si presenta, chiede le generalità dell'utente, nel caso di errori si scusa, oppure nel momento in cui gli vengano fornite delle informazioni ringrazia. Nel caso sia necessario chiede informazioni e chiarimenti e soprattutto non deve scherzare con i dati sensibili degli utenti (Action on Google, 2018; Morrissey et Kirakowski, 2013; Eunji, 2017).

3.22 Empatia e naturalezza

Relazionandosi con gli utenti, l'assistente virtuale deve reagire con moti empatici ad eventuali sentimenti mostrati da essi. Può esternare rabbia, felicità, tristezza in risposta al mood dell'utente (Action on Google, 2018; Eunji, 2017; Fadhil, 2018; Fourault, 2017).

3.23 Originalità

Compito dell'assistente virtuale è saper anche tenere viva la conversazione, quindi può suggerire altri spunti o funzionalità in modo da catturare l'attenzione (Morrissey et Kirakowski, 2013).

3.24 Coerenza

Nel momento in cui si progetta un chatbot deve essere chiaro quale personalità dovrà avere. Quindi se ci si appresta a realizzare un assistente informale potrà muoversi lungo un registro anche piuttosto amichevole, senza cadere in atteggiamenti eccessivamente formali (Action on Google, 2018; Bianchini, 2017; Fadhil, 2018).

4 Risultato dell'analisi

È stata realizzata una stratigrafia [figura 1]: uno studio delle pratiche conosciute fino ad ora, che raccoglie i punti individuati analizzandone le occorrenze, in modo da comprendere quali fra esse siano ormai un'abitudine consolidata e quali, invece, siano tuttora in via di rafforzamento. La stratigrafia vuole rappresentare un sunto rispetto le linee guida incontrate, esplicitando in quante pubblicazioni esse vengono trattate. Accanto ad ogni indicazione viene riportato il numero delle occorrenze. Il compito della stratigrafia è quello di proporre, oltre al mero inventario, anche una riflessione critica rispetto allo stato attuale dello studio intorno alla tematica dei chatbot: non sono stati valutati solo i contributi positivi rispetto a un determinato argomento, ma anche dubbi e problematiche legati ad esso. La prima parte della tabella (1-2 pubblicazioni) indica gli aspetti che sono stati riscontrati una o due volte nell'analisi dei trend di progettazione: alcuni di questi punti sono in realtà fondamentali per il buon design e meriterebbero approfondimenti ulteriori. In particolare, la prima interazione che avviene tra bot e umano è un passaggio importante nell'approccio che gli utenti hanno con gli assistenti virtuali, così come è quasi dato per scontato che la conversazione debba prevedere dei turni (turn taking). La seconda parte della tabella (3-4 pubblicazioni) prende atto delle linee guida in fase di consolidamento per quanto riguarda la letteratura: sono indicazioni per le quali si conta comunque un numero più alto di riferimenti e che sono stati trattati in maniera più approfondita. La terza parte della stratigrafia (5 pubblicazioni e oltre) non rappresenta solo le linee guida più discusse, ma vede trattati alcuni aspetti critici come l'antropomorfizzazione e la sicurezza. In particolare, è stato messo in luce che una rappresentazione troppo umana del bot provochi dei fenomeni di repulsione: tuttavia è necessario che in qualche maniera ci si avvicini a tale raffigurazione, specialmente in un'ottica di conversazione human-like. Inoltre, la sicurezza risulta una delle necessità più importanti per gli utenti: questa esigenza deve essere soddisfatta per ottenere fiducia da parte degli interlocutori. In ogni caso i punti qui presentati sono oggetto di ampia discussione in ambito di design.

1-2 PUBBLICAZIONI
SOLUZIONE INVISIBILE AI PROBLEMI (2)
CONOSCENZA DEI PATH (2)
CAPACITA' DI PREDIZIONE (2)
TURN TAKING (1)
CONVERSAZIONI PRECEDENTI VISIBILI (2)
CHIUDERE LA CONVERSAZIONE IN MODO OPPORTUNO (2)
ORIGINALITA' (1)
3-4 PUBBLICAZIONI
FOCUS SU QUESTIONI SPECIFICHE (3)
RIDUZIONE DEL CARICO COGNITIVO (4)
COMPRENSIONE DEL LINGUAGGIO NATURALE (4)
INPUT VALIDATION/FEEDBACK (4)
CONVERSAZIONI LINEARI CORTE (4)
CONOSCENZA DEL CONTESTO LINGUISTICO (4)
GESTIONE DEGLI ERRORI E WAY OUT (4)
EMPATIA E NATURALEZZA (4)
5 PUBBLICAZIONI E OLTRE
COMPRENSIONE DEL CONTESTO D'USO E DEVICE (6)
ANTROPOMORFIZZAZIONE (6)
SICUREZZA (5)
UTILIZZO DEL MENU A BOTTONI (7)
FLESSIBILITA' (6)

Figura 1 - Stratigrafia

5 Conclusioni

Grazie al lavoro di analisi e ricerca svolto è stato possibile identificare, almeno a livello preliminare, le linee guida utilizzabili in fase di progettazione dei chatbot, specificando quali di queste linee guida siano ancora in fase di discussione e accettazione, e quali invece risultino pratica consolidata per il design di chatbot. Tali linee guida sono in discussione nell'ambito del progetto TIM "Cognitive Solution for Intelligent Caring" (Notiziario Tecnico TIM, 2018) al fine di una loro adozione per garantire una efficace CX.

References

Accenture Interactive, Chatbots in customer service, www.accenture.com

Action on Google, The conversational UI and why it matters, 2018 www.developers.google.com

Amdocs, Human vs Machine: how to stop your virtual agent from lagging behind, 2017 http://solutions.amdocs.com

Theo Araujo, Living up to the chatbot hype: The inlfuence of antropomorphic design cues and communicative agency framing on conversational agent anc company perception, Amsterdam School of Communication Research, University of Amsterdam, in Computers in Human Behaviour 85, 183-185, 2018

Grace M. Begany, Ning Sa, Xiaojun Yuan, Factors affecting user perception of a spoken language vs. textual search interface, Interacting with computers, 28, 2: 170-180, 2015

Alessia Bianchini, BotConference, 2017 www.convcomp.it

Nick C. Bradley, Thomas Fritz,Reid Holmes, Context-Aware Conversational Developer Assistants, in proceedings of 40th International Conference on Software Engineering, Gothenburg, Sweden, (ICSE'18), 2018

Capgemini Digital Transformation Istitute, Conversational Commerce. Why consumers are embracing voice assistant in their life, 2017 www.capgemini.com

Chatbot.org, Consumers say no to Chatbot Silos in US and UK Survey, 2018 www.chatbots.org

Leon Ciechanowski, et al., In the shades of the uncanny valley: An experimental study of human–chatbot interaction, Future Generation Computer Systems, 2018 https://doi.org/10.1016/j.future.2018.01.055.

Florian Daniel, Maristella Matera, Vittorio Zaccaria, and Alessandro Dell'Orto. 2018. Toward Truly Personal Chatbots: On the Development of Custom Conversational Assistants.

Sébastien Fourault, The ultimate guide to designing a chatbot tech stack, 2017 www.chatbotsmagazine.com

Seo Eunji, 11 More best UX practices for building chatbots, 2017 www.chatbotsmagazine.com

Seo Eunji, 19 UX Best practices for building chatbots, 2017 www.chatbotsmagazine.com

Amhed Fadhil, Domain specific design patterns: designing for conversational user interfaces, University of Trento, 2018 https://arxiv.org/ftp/arxiv/papers/1802/1802.09055.pdf

Kim Kangsoo, Luke Boelling, Steffen Haesler, Jeremy N. Bailenson, Gerd Bruder, Gregory F. Welch, Does a Digital Assistant Need a Body? The Influence of Visual Embodiment and Social Behavior on the Perception of Intelligent Virtual Agents in AR, 2018 https://sreal.ucf.edu/wp-content/uploads/2018/08/Kim2018a.pdf

Scott Kevin, Usability heuristics for bots, 2016, www.chatbotsmagazine.com

Dominique Knutsen, Ludovic Le Bigot, Christine Ros, Explicit feedback from users attenuates memory biases in human-system dialogue, J. of Human-Computer Studies 97 (2017) 77–87, 2016

Hanna Limerick, James W. Moore, David Coyle, Empirical Evidence for a Diminished Sense of Agency in Speech Interfaces. Proceedings of ACM Conference on Human factos in Computing system, Seul, Republic of South Corea ACM CHI '15, 2015

Ewa Luger, Abigail Sellen, Like Having a Really Bad PA, Proceedings of 2016 Conference on human factors in Computing System, ACM CHI '16, San Jose, California, 2016

Vinayak Mathur, Arpit Singh, The rapidly changing landscape of conversational agent, College of Information and Computer Sciences, University of Massachusetts Amherst, 2018 https://arxiv.org/pdf/1803.08419.pdf

Microsoft, Bot service documentation, 2018 https://docs.microsoft.com/it-it/azure/bot-service/

Jain Mohit, Kota Ramachandra, Kumar Pratyush, Patel Shwetak, Convey: exploring the use of a context view for chatbots, 2018 https://homes.cs.washington.edu/~mohitj/convey-chi2018.pdf

Kellie Morrissey, Jurek Kirakowski, Realness in chatbots: Establishing quantifiable criteria. In Kurosu M. (eds) Human-Computer Interaction. Interaction Modalities and Techniques. HCI 2013, lecture Notes in Computer Science, vol 8007, Springer, Berlin, Heidelberg, 2013

Meschkat Steffen, Disambiguation of entity references using related entities, Technical Disclosure Commons, 2018 https://www.tdcommons.org/dpubs_series/1108

Francisco Valério, Tatiane Gomes Guimarães, Raquel Prates, Heloisa Candello, Here's what I can do: Chatbots' strategies to convey their features to users, IHC'17, Proceedings of the 16th Brazilian symposium on human factors in computing systems, Joinville, Brazil, 2017

Milan Van Eeuwen, Mobile conversational commerce: messenger chatbots as the next interface

between businesses and consumers, University of Twente http://essay.utwente.nl/71706/1/van%20Eeuwen_MA_BMS.pdf

Mathur Vinayak, Singh Arpit, The rapidly changing landscape of conversational agent, College of Information and Computer Sciences, University of Massachusetts Amherst, 2018 https://arxiv.org/pdf/1803.08419.pdf

Notiziario Tecnico TIM, AI & Customer Interaction http://www.telecomitalia.com/tit/it/notiziariotecnico/edizioni-2018/n-2-2018/capitolo-5.html

Oracle, Can virtual experiences replace reality? The future role for humans in delivering customer experience, 2016 www.oracle.com

Advances in Multiword Expression Identification for the Italian language: The PARSEME shared task edition 1.1

Johanna Monti[1], Silvio Ricardo Cordeiro[2], Carlos Ramisch[2]
Federico Sangati[1], Agata Savary[3], Veronika Vincze[4]
[1]University L'Orientale, Naples, Italy [2]Aix Marseille Univ, CNRS, LIS, Marseille, France
[3]University of Tours, France [4]MTA-SZTE Research Group on Artificial Intelligence, Hungary
jmonti@unior.it,silvioricardoc@gmail.com
carlos.ramisch@lis-lab.fr, fsangati@unior.it
agata.savary@univ-tours.fr, vinczev@inf.u-szeged.hu

Abstract

English. This contribution describes the results of the second edition of the shared task on automatic identification of verbal multiword expressions, organized as part of the *LAW-MWE-CxG 2018 workshop*, co-located with COLING 2018, concerning both the PARSEME-IT corpus and the systems that took part in the task for the Italian language. The paper will focus on the main advances in comparison to the first edition of the task.

Italiano. *Il presente contributo descrive i risultati della seconda edizione dello 'Shared task on automatic identification of verbal multiword expressions' organizzato nell'ambito del LAW-MWE-CxG 2018 workshop realizzato durante il COLING 2018 riguardo sia il corpus PARSEME-IT e i sistemi che hanno preso parte nel task per quel che riguarda l'italiano. L'articolo tratta i principali progressi ottenuti a confronto con la prima edizione del task.*

1 Introduction

Multiword expressions (MWEs) are a particularly challenging linguistic phenomenon to be handled by NLP tools. In recent years, there has been a growing interest in MWEs since the possible improvements of their computational treatment may help overcome one of the main shortcomings of many NLP applications, from Text Analytics to Machine Translation. Recent contributions to this topic, such as Mitkov et al. (2018) and Constant et al. (2017) have highlighted the difficulties that this complex phenomenon, halfway between lexicon and syntax, characterized by idiosyncrasy on various levels, poses to NLP tasks.

This contribution will focus on the advances in the identification of verbal multiword expressions (VMWEs) for the Italian language. In Section 2 we discuss related work. In Section 3 we give an overview of the PARSEME shared task. In Section 4 we present the resources developed for the Italian language, namely the guidelines and the corpus. Section 5 is devoted to the annotation process and the inter-annotator agreement. Section 6 briefly describes the thirteen systems that took part in the shared task and the results obtained. Finally, we discuss conclusions and future work (Section 7).

2 Related work

MWEs have been the focus of the PARSEME COST Action, which enabled the organization of an international and highly multilingual research community (Savary et al., 2015). This community launched in 2017 the first edition of the PARSEME shared task on automatic identification of verbal MWEs, aimed at developing universal terminologies, guidelines and methodologies for 18 languages, including the Italian language (Savary et al., 2017). The task was co-located with the 13th Workshop on Multiword Expressions (MWE 2017), which took place during the European Chapter of the Association for Computational Linguistics (EACL 2017). The main outcomes for the Italian language were the **PARSEME-IT Corpus**, a 427-thousand-word annotated corpus of verbal MWEs in Italian (Monti et al., 2017) and the participation of four systems[1], namely TRANSITION, a transition-based dependency parsing system (Al Saied et al., 2017), SZEGED based on the POS and dependency modules of the Bohnet parser (Simkó et al., 2017), ADAPT (Maldonado et al., 2017) and RACAI (Boroş et al., 2017), both based on sequence la-

[1] http://multiword.sourceforge.net/sharedtaskresults2017

beling with CRFs. Concerning the identification of verbal MWEs some further recent contributions specifically focusing on the Italian language are:

- A supervised token-based identification approach to Italian Verb+Noun expressions that belong to the category of complex predicates (Taslimipoor et al., 2017). The approach investigates the inclusion of concordance as part of the feature set used in supervised classification of MWEs in detecting literal and idiomatic usages of expressions. All concordances of the verbs *fare* ('to do/ to make'), *dare* ('to give'), *prendere* ('to take') and *trovare* ('to find') followed by any noun, taken from the itWaC corpus (Baroni and Kilgarriff, 2006) using SketchEngine (Kilgarriff et al., 2004) are considered.

- A neural network trained to classify and rank idiomatic expressions under constraints of data scarcity (Bizzoni et al., 2017).

With reference to corpora annotated with VMWEs for the Italian language and in comparison with the state of the art described in Monti et al. (2017), there are no further resources available so far. At the time of writing, therefore, the PARSEME-IT VMWE corpus still represents the first sample of a corpus which includes several types of VMWEs, specifically developed to foster NLP applications. The corpus is freely available, with the latest version (1.1) representing an enhanced corpus with some substantial changes in comparison with version 1.0 (cf. Section 4).

3 The PARSEME shared task

The second edition of the PARSEME shared task on automatic identification of verbal multiword expressions (VMWEs) was organized as part of the LAW-MWE-CxG 2018 workshop co-located with COLING 2018 (Santa Fe, USA)[2] and aimed at identifying verbal MWEs in running texts. According to the rules set forth in the shared task, system results could be submitted in two tracks:

- CLOSED TRACK: Systems using only the provided training/development data - VMWE annotations + morpho-syntactic data (if any) - to learn VMWE identification models and/or rules.

- OPEN TRACK: Systems using or not the provided training/development data, plus any additional resources deemed useful (MWE lexicons, symbolic grammars, wordnets, raw corpora, word embeddings, language models trained on external data, etc.). This track includes notably purely symbolic and rule-based systems.

The PARSEME members elaborated for each language i) annotation guidelines based on annotation experiments ii) corpora in which VMWEs are annotated according to the guidelines. Corpora were split in training, development and tests corpora for each language. Manually annotated training and development corpora were made available to the participants in advance, in order to allow them to train their systems and to tune/optimize the systems' parameters. Raw (unannotated) test corpora were used as input to the systems during the evaluation phase. The contribution of the PARSEME-IT research group[3] to the shared task is described in the next section.

4 Italian resources for the shared task

The PARSEME-IT research group contributed to the edition 1.1 of the shared task with the development of specific guidelines for the Italian language and with the annotation of the Italian corpus with over 3,700 VMWEs.

4.1 The shared task guidelines

The 2018 edition of the shared task relied on enhanced and revised guidelines (Ramisch et al., 2018). The guidelines[4] are provided with Italian examples for each category of VMWE.

The guidelines include two universal categories, i.e. valid for all languages participating in the task:

- **Light-verb constructions (LVCs)** with two subcategories: LVCs in which the verb is semantically totally bleached (LVC.full) like in *fare un discorso* ('to give a speech'), and LVCs in which the verb adds a causative meaning to the noun (LVC.cause) like in *dare il mal di testa* ('to give a headache');

- **Verbal idioms (VIDs)** like *gettare le perle ai porci* ('to throw pearls before swine').

[2] https:http://multiword.sourceforge.net/lawmwecxg2018

[3] https://sites.google.com/view/parseme-it/home

[4] http://parsemefr.lif.univ-mrs.fr/parseme-st-guidelines/1.1/

Three quasi-universal categories, valid for some language groups or languages but non-existent or very exceptional in others are:

- **Inherently reflexive verbs (IRV)** which are those reflexive verbal constructions which (a) never occur without the clitic e.g. *suicidarsi* ('to suicide'), or when (b) the IRV and non-reflexive versions have clearly different senses or subcategorization frames e.g. *riferirsi* ('to refer') opposed to *riferire* ('to report / to tell');

- **Verb-particle constructions (VPC)** with two subcategories: fully non-compositional VPCs (VPC.full), in which the particle totally changes the meaning of the verb, like *buttare giù* ('to swallow') and semi non-compositional VPCs (VPC.semi), in which the particle adds a partly predictable but non-spatial meaning to the verb like in *andare avanti* ('to proceed');

- **Multi-verb constructions (MVC)** composed by a sequence of two adjacent verbs like in *lasciar perdere* ('to give up').

An optional experimental category (if admitted by the given language, as is the case for Italian) is considered in a post-annotation step:

- **Inherently adpositional verbs (IAVs)**, which consist of a verb or VMWE and an idiomatic selected preposition or postposition that is either always required or, if absent, changes the meaning of the verb significantly, like in *confidare su* ('to trust on').

Finally, a language-specific category was introduced for the Italian language:

- **Inherently clitic verbs (LS.ICV)** formed by a full verb combined with one or more non-reflexive clitics that represent the pronominalization of one or more complements (CLI). LS.ICV is annotated when (a) the verb never occurs without one non-reflexive clitic, like in *entrarci* ('to be relevant to something'), or (b) when the LS.ICV and the non-clitic versions have clearly different senses or subcategorization frames like in *prenderle* ('to be beaten') vs *prendere* ('to take').

4.2 The PARSEME-IT corpus

The PARSEME-IT VMWE corpus version 1.1 is an updated version of the corpus used for edition 1.0 of the shared task. It is based on a selection of texts from the PAISÀ corpus of web texts (Lyding et al., 2014), including Wikibooks, Wikinews, Wikiversity, and blog services. The PARSEME-IT VMWE corpus was updated in edition 1.1 according to the new guidelines described in the previous section. Table 4.2 summarizes the size of the corpus developed for the Italian language and presents the distribution of the annotated VMWEs per category.

The training, development and test data are available in the LINDAT/Clarin repository[5], and all VMWE annotations are available under Creative Commons licenses (see README.md files for details). The released corpus' format is based on an extension of the widely-used CoNLL-U file format.[6]

5 Annotation process

The annotation was manually performed in running texts using the FoLiA linguistic annotation tool[7] (van Gompel and Reynaert, 2013) by six Italian native speakers with a background in linguistics, using a specific decision tree for the Italian language for joint VMWE identification and classification.[8]

In order to allow the annotation of IAVs, a new pre-processing step was introduced to split compound prepositions such as *della* ('of the') into two tokens. This step was necessary to annotate only lexicalised components of the IAV, as in *portare alla disperazione*, where only the verb and the preposition *a* should be annotated, without the article *la*.

Once the annotation was completed, in order to reduce noise and to increase the consistency of the annotations, we applied the consistency checking tool developed for edition 1.0 (Savary et al., forthcoming). The tool groups all annotations of the same VMWE, making it possible to spot annotation inconsistencies very easily.

[5] http://hdl.handle.net/11372/LRT-2842

[6] http://multiword.sf.net/cupt-format

[7] http://mwe.phil.hhu.de/

[8] http://parsemefr.lif.univ-mrs.fr/parseme-st-guidelines/1.1/?page=it-dectree

	sent.	tokens	VMWEs	IAV	IRV	LS.ICV	LVC.cause/full	MVC	VID	VPC.full/semi
IT-dev	917	32613	500	44	106	9	19/100	6	197	17/2
IT-train	13555	360883	3254	414	942	20	147/544	23	1098	66/0
IT-test	1256	37293	503	41	96	8	25/104	5	201	23/0
IT-Total	15728	430789	4257	499	7641	37	191/748	35	1496	106/2

Table 1: Statistics of the PARSEME-IT corpus version 1.1.

	#S	#A_1	#A_2	F$_{span}$	κ_{span}	κ_{cat}
PARSEME-IT-2017	2000	336	316	0.417	0.331	0.78
PARSEME-IT-2018	1000	341	379	0.586	0.550	0.882

Table 2: IAA scores for the PARSEME-IT corpus in versions from 2017 and 2018: #S is the number of sentences in the double-annotated corpus used for measuring the IAA. #A_1 and #A_2 refer to the number of VMWE instances annotated by each of the annotators. F_{span} is the F-measure for identifying the span of a VMWE, when considering that one of the annotators tries to predict the other's annotations (VMWE categories are ignored). κ_{span} and κ_{cat} are the values of Cohen's κ for span identification and categorization, respectively.

5.1 Inter-annotator agreement

A small portion of the corpus consisting in 1,000 sentences was double-annotated. In comparison with the previous edition, the inter-annotator agreement shown in Table 2 increased, although it is still not optimal.[9] The improvement is probably due to the fact that, this time, the group was based in one place with the exception of one annotator, and several meetings took place prior to the annotation phase in order to discuss the new guidelines.

The two annotators involved in the IAA task annotated 191 VMWEs with no disagreement, but there were several problems, which led to 44 cases of partial disagreement and 250 cases of total disagreement:

- PARTIAL MATCHES LABELED, (25 cases) in which there is at least one token of the VMWE in common between two annotators and the labels assigned are the same. The disagreement mainly concerns the lexicalized elements as part of the VMWE, as in the case of the VID *porre in cattiva luce* ('make look bad'). Annotators disagreed, indeed, about considering the adjective *cattiva* ('bad') as part of the VID.
- EXACT MATCHES UNLABELED, (18 cases) in which the annotators agreed on the lexicalized components of the VMWE to be annotated but not the label. This type of disagreement is mainly related to fine-grained categories such as **LVC.cause** and **LVC.full** as in the case of *dare ... segnale* (to give ... a signal) or **VPC.full** and **VPC.semi** as for *mettere insieme* ('to put together')
- PARTIAL MATCHES UNLABELED, (1 case) in which there is at least one token of the VMWE in common between two annotators but the labels assigned are different, such as in *buttar-si in la calca* ('to join the crowd') classified as VID by the first annotator and *buttar-si* ('to throw oneself') classified as IRV by the second one in the following sentence: *[...] attendendo il venerdì sera per buttarsi nella calca del divertimento [...].* ('waiting for the Friday evening to join the crowd for entertainment')
- ANNOTATIONS CARRIED OUT ONLY BY ONE OF THE ANNOTATORS: This is the category which collects the most numerous examples of disagremeent between annotators: 106 VMWE were annotated only by annotator 1 and 144 by annotator 2.

6 The systems and the results of the shared task for the Italian language

Whereas only four systems took part in edition 1.0 of the shared task for the Italian language, in edition 1.1, fourteen systems took on this challenge. The system that took part in the PARSEME shared task are listed in Table 3: 12 took part in the closed track and two in the open one. The two systems that took part in the open track reported the resources that were used, namely SHOMA used pretrained wikipedia word embeddings (Taslimipoor and Rohanian, 2018), while Deep-BGT (Berk et al., 2018) relied on the BIO tagging scheme

[9] As mentioned in Ramisch et al. (2018), the estimation of chance agreement in κ_{span} and κ_{cat} is slightly different between 2017 and 2018, therefore these results are not directly comparable.

and its variants (Schneider et al., 2014) to introduce additional tags to encode gappy (discontinuous) VMWEs. A distinctive characteristic of the systems of edition 1.1 is that most of them (GBD-NER-resplit and GBD-NER-standard, TRAPACC, and TRAPACC-S, SHOMA, Deep-BGT) use neural networks, while the rest of the systems adopt other approaches: CRF-DepTree-categs and CRF-Seq-nocategs are based on a tree-structured CRF, MWETreeC and TRAVERSAL on syntactic trees and parsing methods, Polirem-basic and Polirem-rich on statistical methods and association measures, and finally varIDE uses a Naive Bayes classifier. The systems were ranked according two types of evaluation measures (Ramisch et al., 2018): a strict per-VMWE score (in which each VMWE in gold is either deemed predicted or not, in a binary fashion) and a fuzzy per-token score (which takes partial matches into account). For each of these two, precision (P), recall (R) and F1-scores (F) were calculated. Table 3 shows the ranking of the systems which participated in the shared task for the Italian language. The systems with highest MWE-based Rank for Italian have F1 scores that are mostly comparable to the scores obtained in the General ranking of all languages (e.g. TRAVERSAL had a General F1 of 54.0 vs Italian F1 of 49.2, being ranked first in both cases). Nevertheless, the Italian scores are consistently lower than the ones in the General ranking, even if only by a moderate margin, suggesting that Italian VMWEs in this specific corpus might be particularly harder to identify. One of the outliers in the table is MWETreeC, which predicts much fewer VMWEs than in the annotated corpora. This turned out to be true for other languages as well. The few VMWEs that were predicted only obtained partial matches, which explains why its MWE-based score was 0. Another clear outlier is Polirem-basic. Both Polirem-basic and Polirem-rich had predictions for Italian, French and Portuguese. Their scores are somewhat comparable in the three languages, suggesting that the lower scores are a characteristic of the system and not some artifact of the Italian corpus.

TRASVERSAL (Waszczuk, 2018) was the best performing system in the closed track, while SHOMA (Taslimipoor and Rohanian, 2018) performed best in the open one. As shown in Figure 1, comparing the MWE-based F1 scores for each label for the two best performing systems,

System	Track	MWE-based				Token-based			
		P	R	F1	Rank	P	R	F1	Rank
TRAVERSAL	closed	63.09	40.32	**49.2**	1	74.42	42.11	**53.78**	1
TRAPACC	closed	52.43	30.44	**38.52**	2	61.54	30.34	**40.64**	4
CRF-Seq-nocategs	closed	55.14	27.02	**36.27**	3	78.49	33.05	**46.51**	2
TRAPACC_S	closed	55.66	23.79	**33.33**	4	65.42	22.99	**34.02**	8
CRF-DepTree-categs	closed	44.76	25.81	**32.74**	5	58.78	29.8	**39.55**	5
varIDE	closed	31.07	34.07	**32.5**	6	39.22	35.06	**37.02**	6
Veyn	closed	34.01	30.44	**32.13**	7	58.41	38.16	**46.16**	3
Polirem-rich	closed	72.36	17.94	**28.76**	8	86.54	21.9	**34.96**	7
GBD-NER-standard	closed	15.45	29.84	**20.36**	9	22.68	35.45	**27.67**	9
GBD-NER-resplit	closed	10.69	28.83	**15.59**	10	16.63	38.31	**23.19**	10
Polirem-basic	closed	83.33	4.03	**7.69**	11	81.82	3.48	**6.68**	11
MWETreeC	closed	0	0	**0**	n/a	1.45	6.58	**2.38**	12
SHOMA	open	50.37	41.33	**45.4**	1	67.49	46.59	**55.13**	1
Deep-BGT	open	45.52	25.6	**32.77**	2	70	27.63	**39.62**	2

Table 3: Results for the Italian language

TRASVERSAL obtained overall better results for almost all VMWEs categories with the exception of VID and MVC, for which SHOMA showed a better performance.

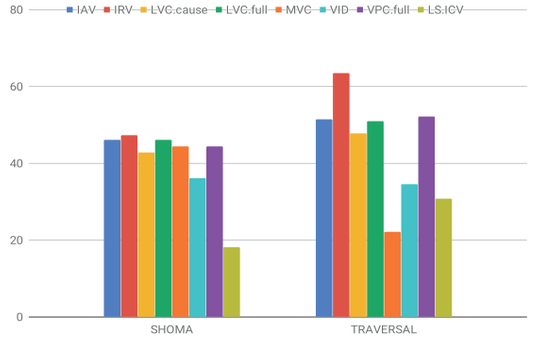


Figure 1: Chart comparing the MWE-based F1 scores for each label of the two best performing systems.

7 Conclusions and future work

Having presented the results of the PARSEME shared task edition 1.1, the paper described the advances achieved in this last edition in comparison with the previous one, but also highlighted that there is room for further improvements. We are working on some critical areas which emerged during the annotation task in particular with reference to some borderline cases and the refinement of the guidelines. Future work will focus on maintaining and increasing the quality and the size of the corpus but also on extending the shared task to other MWE categories, such as nominal MWEs.

Acknowledgments

Our thanks go to the Italian annotators Valeria Caruso, Maria Pia di Buono, Antonio Pascucci, Annalisa Raffone, Anna Riccio for their contributions.

References

Hazem Al Saied, Matthieu Constant, and Marie Candito. 2017. The ATILF-LLF system for PARSEME shared task: a transition-based verbal multiword expression tagger. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 127–132.

Marco Baroni and Adam Kilgarriff. 2006. Large linguistically-processed web corpora for multiple languages. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations*, pages 87–90. Association for Computational Linguistics.

Gözde Berk, Berna Erden, and Tunga Güngör. 2018. Deep-bgt at parseme shared task 2018: Bidirectional lstm-crf model for verbal multiword expression identification. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 248–253.

Yuri Bizzoni, Marco S. G. Senaldi, and Alessandro Lenci. 2017. Deep-learning the Ropes: Modeling Idiomaticity with Neural Networks. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017), Rome, Italy, December 11-13, 2017.*

Tiberiu Boroş, Sonia Pipa, Verginica Barbu Mititelu, and Dan Tufiş. 2017. A data-driven approach to verbal multiword expression detection. PARSEME Shared Task system description paper. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 121–126.

Mathieu Constant, Gülşen Eryiğit, Johanna Monti, Lonneke van der Plas, Carlos Ramisch, Michael Rosner, and Amalia Todirascu. 2017. Multiword Expression Processing: A Survey. *Computational Linguistics*, 43(4):837–892. URL `https://doi.org/10.1162/COLI_a_00302`.

Adam Kilgarriff, Pavel Rychly, Pavel Smrz, and David Tugwell. 2004. Itri-04-08 the sketch engine. *Information Technology*, 105:116.

Verena Lyding, Egon Stemle, Claudia Borghetti, Marco Brunello, Sara Castagnoli, Felice Dell'Orletta, Henrik Dittmann, Alessandro Lenci, and Vito Pirrelli. 2014. The PAISÀ Corpus of Italian Web Texts. In *Proceedings of the 9th Web as Corpus Workshop (WaC-9)*, pages 36–43. Association for Computational Linguistics, Gothenburg, Sweden. URL `http://www.aclweb.org/anthology/W14-0406`.

Alfredo Maldonado, Lifeng Han, Erwan Moreau, Ashjan Alsulaimani, Koel Chowdhury, Carl Vogel, and Qun Liu. 2017. Detection of Verbal Multi-Word Expressions via Conditional Random Fields with Syntactic Dependency Features and Semantic Re-Ranking. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 114–120. Association for Computational Linguistics.

Ruslan Mitkov, Johanna Monti, Gloria Corpas Pastor, and Violeta Seretan. 2018. *Multiword units in machine translation and translation technology*, volume 341. John Benjamins Publishing Company.

Johanna Monti, Maria Pia di Buono, and Federico Sangati. 2017. PARSEME-IT Corpus. An annotated Corpus of Verbal Multiword Expressions in Italian. In *Fourth Italian Conference on Computational Linguistics-CLiC-it 2017*, pages 228–233. Accademia University Press.

Carlos Ramisch, Silvio Ricardo Cordeiro, Agata Savary, Veronika Vincze, Verginica Barbu Mititelu, Archna Bhatia, Maja Buljan, Marie Candito, Polona Gantar, Voula Giouli, Tunga Güngör, Abdelati Hawwari, Uxoa Iñurrieta, Jolanta Kovalevskaitė, Simon Krek, Timm Lichte, Chaya Liebeskind, Johanna Monti, Carla Parra Escartín, Behrang QasemiZadeh, Renata Ramisch, Nathan Schneider, Ivelina Stoyanova, Ashwini Vaidya, and Abigail Walsh. 2018. Edition 1.1 of the PARSEME Shared Task
on Automatic Identification of Verbal Multiword Expressions. In *the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 222–240.

Agata Savary, Marie Candito, Verginica Barbu Mi-

titelu, Eduard Bejček, Fabienne Cap, Slavomír Čéplö, Silvio Ricardo Cordeiro, Gülşen Eryiğit, Voula Giouli, Maarten van Gompel, Yaakov HaCohen-Kerner, Jolanta Kovalevskaitė, Simon Krek, Chaya Liebes kind, Johanna Monti, Carla Parra Escartín, Lonneke van der Plas, Behrang QasemiZadeh, Carlos Ramisch, Federico Sangati, Ivelina Stoyanova, and Veronika Vincze. forthcoming. PARSEME multilingual corpus of verbal multiword expressions. In Stella Markantonatou, Carlos Ramisch, Agata Savary, and Veronika Vincze, editors, *Multiword expressions at length and in depth. Extended papers from the MWE 2017 workshop*. Language Science Press, Berlin, Germany.

Agata Savary, Carlos Ramisch, Silvio Cordeiro, Federico Sangati, Veronika Vincze, Behrang QasemiZadeh, Marie Candito, Fabienne Cap, Voula Giouli, Ivelina Stoyanova, and Antoine Doucet. 2017. The parseme shared task on automatic identification of verbal multiword expressions. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 31–47. Association for Computational Linguistics, Valencia, Spain. URL http://www.aclweb.org/anthology/W17-1704.

Agata Savary, Manfred Sailer, Yannick Parmentier, Michael Rosner, Victoria Rosén, Adam Przepiórkowski, Cvetana Krstev, Veronika Vincze, Beata Wójtowicz, Gyri Smørdal Losnegaard, Carla Parra Escartín, Jakub Waszczuk, Mathieu Constant, Petya Osenova, and Federico Sangati. 2015. PARSEME – PARSing and Multiword Expressions within a European multilingual network. In *7th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics (LTC 2015)*. Poznań, Poland. URL https://hal.archives-ouvertes.fr/hal-01223349.

Nathan Schneider, Emily Danchik, Chris Dyer, and Noah A Smith. 2014. Discriminative lexical semantic segmentation with gaps: running the mwe gamut. *Transactions of the Association for Computational Linguistics*, 2:193–206.

Katalin Ilona Simkó, Viktória Kovács, and Veronika Vincze. 2017. USzeged: Identifying Verbal Multiword Expressions with POS Tagging and Parsing Techniques. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 48–53.

Shiva Taslimipoor and Omid Rohanian. 2018. Shoma at parseme shared task on automatic identification of vmwes: Neural multiword expression tagging with high generalisation. *arXiv preprint arXiv:1809.03056*.

Shiva Taslimipoor, Omid Rohanian, Ruslan Mitkov, and Afsaneh Fazly. 2017. Investigating the Opacity of Verb-Noun Multiword Expression Usages in Context. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 133–138. Association for Computational Linguistics. URL http://aclweb.org/anthology/W17-1718.

Maarten van Gompel and Martin Reynaert. 2013. FoLiA: A practical XML Format for Linguistic Annotation-a descriptive and comparative study. *Computational Linguistics in the Netherlands Journal*, 3:63–81.

Jakub Waszczuk. 2018. Traversal at parseme shared task 2018: Identification of verbal multiword expressions using a discriminative tree-structured model. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 275–282.

PARSEME-IT
Issues in verbal Multiword Expressions identification and classification

Johanna Monti[1], **Valeria Caruso**[1], **Maria Pia di Buono**[2]

[1]Dep. of Literary, Linguistic and Comparative Studies "L'Orientale" University of Naples, Italy
[2]TakeLab - Faculty of Electrical Engineering and Computing - University of Zagreb, Croatia
jmonti@unior.it, vcaruso@unior.it, mariapia.dibuono@fer.hr

Abstract

English. The second edition of the PARSEME shared task was based on new guidelines and methodologies that particularly concerned the Italian language with the introduction of new categories of verbs not considered in the previous edition. This contribution presents the novelties introduced, the results obtained and the problems that emerged during the annotation process and concerning some categories of verbs.

Italiano. *La seconda edizione del PARSEME shared task si è basata su nuove linee guida e metodologie che hanno riguardato in particolar modo la lingua italiana con l'introduzione di nuove categorie di verbi non considerate nella precedente edizione. Il contributo presenta le novità introdotte, i risultati ottenuti e le problematiche che sono emerse durante l'annotazione relativamente ad alcune categorie di verbi.*

1 Introduction

The paper reports on some final results of the second edition of an annotation trial for verbal Multiword Expressions (VMWEs) carried out on the Italian language by the PARSEME-IT research group [1], which started within the broader European PARSEME project, the IC1207 COST action ended in April 2017[2].

The initial project is expanding in this second stage of its development, thanks to a wider network of research groups, working together as one of the ACL Special Interest Group on the Lexicon, called SIGLEX-MWE.

In its first edition, the PARSEME shared task released a corpus of 5.5 million tokens and 60,000 VMWE annotations in 18 different languages which is distributed under different versions of the Creative Commons license. To increase the computational efficiency of Natural Language Processing (NLP) applications, PARSEME focuses on a special class of Multiword Expressions which have been seldom modelled for their challenging nature, such as verbal MWEs (Savary et al., 2017).

Many of the features of this particular type of MWE are considered to be difficult to cope with, such as the discontinuity they present (*turn it off*) the syntactic variations they license (*the decision was hard to take*), the semantic variability resulting both in literal and idiomatic readings (*to take the cake*), or the syntactic ambiguity of many forms (*on* is a preposition in *to trust on somebody*, but a particle in *to take on the task*). Moreover, these units have language-specific features, and are generally modelled according to descriptive categories developed by different traditions of linguistic studies. The PARSEME research group thus addresses also the creation of a multilingual common platform for VMVEs using universal terminology, guidelines and methodologies for the identification of these units cross-linguistically. Moreover, at the end of the first annotation trial a shared task on automatic identification of VMWEs was also carried out and has proved the reliability and usefulness of the data collected so far, which have been already presented and discussed (Savary et al., 2017; Monti et al., 2017).

The paper illustrates the types of VMWEs used by the second PARSEME annotation trial more thoroughly. In Section 2 we provide a brief description of the second annotation trial of the PARSEME shared task together with the statistics. Then we present a new category of verbal MWEs,

[1]https://sites.google.com/view/parseme-it/home
[2]https://typo.uni-konstanz.de/parseme/

namely Inherently Clitic Verbs (Section 3) and in Section 4 two very productive categories in Italian (IRV and IDV). In Section 5, we discuss some borderline cases which posed some classification issues. Finally, we conclude and discuss future work.

2 PARSEME Shared Task Second annotation trial: a brief report

This section focuses on the novelties which have been introduced in the guidelines and methodologies used for the second annotation trial in order to cover a wider range of VMWEs which were left apart in the first stage of the project. The improvements seem to be particularly valuable for the data collection carried out on the Italian language, because they address some peculiarities of the Italian language which were not considered in the first edition of the shared task but have been taken into account in the second edition, namely:

- **Inherently clitic verbs (ICV)**, which is an extremely rich and varied VMWE category in Italian (Masini, 2015). As described in Section 3, a language specific category was created for the Italian language (LS.ICV) which takes into account only those verbs whose semantics is changed by a non-reflexive clitic pronoun, like *entrarci* when it means *to be relevant to something*, while the intranstive form of the verb *entrare* means *to enter*.

- **Inherently adpositional verbs (IAV)**, a high frequency category of VMWEs, namely those verbs whose meanings are significantly affected by an "idiomatic selected preposition", like *su* in *contare su qualcuno* (to rely on someone): without the preposition the verb means only *to determine the total number of something*. These verbs are often called *prepositional verbs*[3].

- **Multi verb constructions (MVC)**, VMWEs composed by a sequence of two adjacent verbs (in a language-dependent order), a governing verb V_gov (also called a vector verb) and a dependent verb V_dep (also called a polar verb), like in *lasciar perdere* (to give up).

[3]Schneider, N., Green, M., 2015, New Guidelines for Annotating Prepositional Verbs, https://github.com/nschneid/nanni/wiki/Prepositional-Verb-Annotation-Guidelines

The other classifying categories used are (a) **light verb constructions (LVCs)**, e.g. *fare una passeggiata* (to have a walk), and (b) **idioms (ID)**, e.g., *tirare le cuoia* (to kick the bucket), considered to be universal categories or categories which can be found in all languages participating in the task.

Other VMWEs are instead maintained as quasi-universal categories, since their range of application seems to cover only some language groups or languages, but not all. They are (c) **inherently reflexive verbs (IReflVs)**, and (d) **verb-particle constructions (VPCs)**. The first group (IReflVs) allows annotators to account for verbs which are never used without a reflexive clitic pronoun, e.g., (Italian) *suicidarsi* (to suicide), or for those verbs whose meaning is significantly affected by the pronoun, e.g., (Italian) *farsi* (to take drugs) while the non-pronominal form, *fare*, means *to make*. Semantic aspects are also used to identify Verb-particle constructions (VPC) because their meaning is fully non-compositional, e.g., *buttare giù* (to swallow), or only partly non-compositional, like in *tirare avanti* (to go on) since the preposition no longer owns its spatial meaning.

Table 1 presents the statistics of the various categories of VMWEs in the PARSEME-IT corpus 1.1.

3 A language specific category: Inherently clitic verbs (LS.ICV)

Inherently Clitic Verbs (LS.ICV) represent a specific category for some Romance languages, and they are particularly frequent in the Italian language. It is often challenging to distinguish LS.ICV from Inherently Reflexive Verbs (IRV), particularly because some clitics may be ambiguous, like *se/si* which is a polyfunctional clitic pronoun and grammatical marker (and can have a reflexive, reciprocal, impersonal, passivizing, aspectual, and middle function). LS.ICVs together with IRVs are pronominal verbs. LS.ICV are formed by a full verb combined with one or more non-reflexive clitic that represents the pronominalization of one or more complement (CLI).

The following verbs should be annotated as LS.ICV:

- The verb without the CLI does not exist, e.g., *infischiarsene* (do not worry about) vs **infischiare*;

	sent.	tokens	VMWEs	IAV	IRV	LS.ICV	LVC.cause/full	MVC	VID	VPC.full/semi
IT-dev	917	32613	500	44	106	9	19/100	6	197	17/2
IT-train	13555	360883	3254	414	942	20	147/544	23	1098	66/0

Table 1: PARSEME-IT corpus version 1.1

- The verb without the CLI does exist, but has a very different meaning as in *prenderle* (gl.: to take them, transl. to be beaten) vs *prendere* (to take) or *prenderci* (gl.: to take it, transl. to grasp the truth) vs *prendere* (to take);
- The verb has more than one CLI of which the second one is an invariable object complement, like in *fregarsene* (gl.: matter self of-it, transl. do not care about) or *infischiarsene* (do not worry about);
- The verb has two non-reflexive invariable CLIs, like in *farcela* (gl.: to make there it, transl. to succeed);
- The verb has a different meaning with respect to an intensive use of the same two non-reflexive invariable CLIs, like in *andarsene* (gl.: to go away self from-there, transl. to die) vs *andarsene* (to go away) or *bersela* (gl.: drink self it, transl. to believe) vs *bersela* (to drink it).

The annotation of LS.ICV was performed following a specific decision tree [4].

In the training corpus 20 different LS.ICV were annotated manually, such as *farcela*, *rimetterci*, *fregarsene* among others.

4 Very productive VMWEs: IRVs and VID

IRVs and VID represent very productive categories in Italian which pose some classifying issues due to their specific characteristics.

With reference to IRVs, the presence of the clitic pronoun *si* may generate ambiguity in the annotation process, as in Italian it refers to three different types of construction: i) reflexive, ii) impersonal, iii) inherent.

In order to distinguish these cases, we consider that in the reflexive construction, the clitic pronoun can be paraphrased by means of either an anaphoric expression which stands for *se stesso* (oneself) or a mutual expression which refers to *gli uni e gli altri* (these and those). Another relevant aspect to consider in the classification of IRVs is the presence of an implicit thematic role due to the fact that the action includes two different entities with different thematic properties but with the same reference, e.g., in *guardarsi* (to look at oneself) the clitic signals the presence of coreference between the first argument and the second one. Another source of mis-classification of IRVs is related to the presence of anticausative constructions. In these constructions, the clitic may represent an overt marker of reduced transitivity, , e.g., *sedersi* (to sit down).

In some cases, IRVs occur in idiomatic construction and their meaning is affected by the presence of new elements, such as in *guardarsi bene da* (to be careful not to). Consequently the annotation of such occurrences is subjected to the evaluation of characteristics related to VID, as the low variability, the presence of semantic non-compositional meaning, and the literal-idiomatic ambiguity. In the VID class, the non-compositionality property is prototypical such as in *battersi all'ultimo sangue* (lit. to fight till the last blood) which means *to fight to the last*. Despite their meaning is opaque, sometimes VID may have both a literal and idiomatic meaning and the boundaries between them are difficult to trace. For example, *avere gli occhi bendati* (lit. to have the eyes covered) has both a literal meaning and an idiomatic one and in this latter case it should be translated in English as *to be blindfold*. According to Vietri (2014b), it is possible to classify ordinary-verb VID, namely VID which present a semantically full verb, on the basis of their definitional structure, identified by means of the arguments required by the operators. In the case of VID, the operator consists of the verb and the fixed element(s), while the argument may be the subject and/or a free complement. VIDs can be formed also by constructions based on the use of support verbs, namely *avere* (to have), e.g., *avere fegato* (lit. to have leaver, transl. to have guts) *essere*

[4] http://parsemefr.lif.univ-mrs.fr/parseme-st-guidelines/1.1/?page=060_Language-specific_tests/015_Inherently_clitic_verbs__LB_LS.ICV_RB_

(to be), e.g., *essere a cavallo* (to be golden) and *fare* (to make), e.g., *fare lo gnorri* (to play fool). The main difference between this class of VID and the one formed by ordinary verbs is that support verbs are semantically empty, and for this reason this class of VID presents a high degree of lexical and syntactic variability. This type of variability is retrievable in aspectual variants, the production of causative constructions, the possible deletion of the support verb which causes complex nominalizations (Vietri, 2014a).

5 Borderline cases: LVC and IAV compared

In this section we discuss the novelties concerning two categories used in the second edition of the PARSEME shared identification task of verbal MWEs (edition 1.1), namely LVC and IAV. As regards LVC, two new subcategories have been introduced in the second edition, LVC.full and LVC.cause, to account for a more fine-grained distinction between LVCs, where the verb is semantically totally bleached (e.g., *to have the right*), and those where the verb adds a causative meaning (and a new semantic role) to the noun (e.g., *to grant the right*). Therefore some new tests have been added to account for these subcategories, which heavily rely on the notion of semantic arguments.

In particular, constructions annotated as LVC.cause may involve: i) verbs that are typically used to express the cause of predicative nouns in general (e.g., *cause, provoke*), ii) verbs that are only used to express the cause of particular predicative nouns (e.g., *grant* in *to grant a right*).

IAV consists of a verb or VMWE and an idiomatic selected preposition or postposition that is either always required or, if absent, changes the meaning of the verb of the VMWE significantly. IAVs are verb+adposition combinations in which: i) the dependents of the adposition are not lexicalized, or ii) the adposition cannot be omitted without markedly altering the meaning of the verb. During the annotation trial, the IAV category has proved to be advantageous to cover the rich inventory of VMWEs in Italian, but some issues have also emerged, particularly with respect to the other class of LVC verbs, which also accounts for combinations of verbs plus prepositions. Prototypical examples of IAV collected so far include the following:

1.a *Tendere a* + N (to be inclined to something), base form *tendere* (to stretch), e.g., *Maria tende alla depressione* (Maria tends to be depressed);

1.b *Tendere a* + V (to be inclined to something), e.g., *Maria tende a dimagrire* (Maria tends to loose weight);

2. *Puntare su* + N (to bet), base form *puntare* (to stick), e.g., *puntare su qualcuno/qualcosa*.

These examples exhibit clear semantic changes from the non-adpositional base form of the verb; moreover, the preposition can not be omitted in questions, thus proving to be part of the verb:

- *Maria tende sempre ad esagerare.*
- *A cosa tende, scusa?*

Less prototypical IAV examples include verb instances exhibiting semantic changes pivoted by the arguments they combine with, like *andare in* (both *to go to* and *to become*), or *sapere di* (*to smell* and *to know about*). The type of semantic interaction at stake, called *co-composition* in the Generative Lexicon[5], is realized when "the complements carry information which acts on the governing verb, essentially taking the verb as argument and shifting its event type" (Pustejovsky, 1995). For example, *andare in* denotes directed motion when combined with proper or common place nouns like in *andare in città/montagna/America*, (to go to the city/mountain/America); or the medium of motion, when combined with vehicles names, like in *vado in bici/Ferrari*, (I ride my bike/drive my Ferrari). However, with nouns denoting *states*, like *andare in estasi* (to become absorbed) or *andare in panico* (to start feel panic), the verb acquires the aspectual meaning of *to go into the state X*, and can not be classified as an LVC. With names referring to events, instead, like *andare in soccorso* (lit. to go in assistance), the original spatial semantics bleaches by interacting with the name meaning: actually *to go into the event X* denotes the action expressed by the predicative name and can be classified as an LVC.

[5]Co-composition has been called '*accommodation* in more recent works (Pustejovsky, 2013).

6 Conclusions and Future Work

In this paper we described the novelties concerning the PARSEME shared task on automatic identification of verbal MWEs - edition 1.1 (2018), in which new verb categories have been included in comparison with the 2017 edition. Some of them are language-specific, such as ICV for some Romance languages, others are not, like IAV. The increased number of categories enables to annotate corpus data more thoroughly, and discover a broad range of combinatorial phenomena that present different degrees of opacity.

We also discussed two productive categories in Italian, namely IRV and VID, and analyzed LVC and IAV borderline cases together with observations on combinatorial phenomena that can be applied in order to annotate VMWE more effectively.

Future work includes a further linguistic analysis of borderline cases in order to contribute to the description of these phenomena.

Acknowledgments

This research has been partly supported by the European Regional Development Fund under the grant KK.01.1.1.01.0009 (DATACROSS).
Authorship contribution is as follows: Johanna Monti is author of Sections 1, 2, and 3; Valeria Caruso of Section 5, and Maria Pia di Buono of Sections 4 and 6.

References

Francesca Masini. 2015. Idiomatic verb-clitic constructions: lexicalization and productivity. In *Mediterranean Morphology Meetings*, volume 9, pages 88–104.

Johanna Monti, Maria Pia di Buono, and Federico Sangati. 2017. Parseme-it corpus an annotated corpus of verbal multiword expressions in italian. In *Fourth Italian Conference on Computational Linguistics-CLiC-it 2017*, pages 228–233. Accademia University Press.

James Pustejovsky. 1995. *The generative lexicon*. MIT Press.

James Pustejovsky. 2013. Type theory and lexical decomposition. In *Advances in generative lexicon theory*, pages 9–38. Springer.

Agata Savary, Carlos Ramisch, Silvio Cordeiro, Federico Sangati, Veronika Vincze, Behrang Qasemizadeh, Marie Candito, Fabienne Cap, Voula Giouli, Ivelina Stoyanova, et al. 2017. The parseme shared task on automatic identification of verbal multiword expressions. In *Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017)*, pages 31–47.

Simonetta Vietri. 2014a. *Idiomatic Constructions in Italian: A Lexicon-grammar Approach*, volume 31. John Benjamins Publishing Company.

Simonetta Vietri. 2014b. The lexicon-grammar of italian idioms. In *Workshop on Lexical and Grammatical Resources for Language Processing, COLING 2014*, pages 137–146.

Local associations and semantic ties in overt and masked semantic priming

Andrea Nadalini
International School for Advanced Studies
Trieste, Italy
anadalini@sissa.it

Marco Marelli
Bicocca University
Milan, Italy
marco.marelli@unimib.it

Roberto Bottini
Center for mind/brain sciences
Trento, Italy
roberto.bottini@unitn.it

Davide Crepaldi
International School for Advanced Studies
Trieste, Italy
dcrepaldi@sissa.it

Abstract

English. Distributional semantic models (DSM) are widely used in psycholinguistic research to automatically assess the degree of semantic relatedness between words. Model estimates strongly correlate with human similarity judgements and offer a tool to successfully predict a wide range of language-related phenomena. In the present study, we compare the state-of-art model with pointwise mutual information (PMI), a measure of local association between words based on their surface cooccurrence. In particular, we test how the two indexes perform on a dataset of sematic priming data, showing how PMI outperforms DSM in the fit to the behavioral data. According to our result, what has been traditionally thought of as semantic effects may mostly rely on local associations based on word co-occurrence.

Italiano. *I modelli semantici distribuzionali sono ampiamente utilizzati in psicolinguistica per quantificare il grado di similarità tra parole. Tali stime sono in linea con i corrispettivi giudizi umani, e offrono uno strumento per modellare un'ampia gamma di fenomeni relativi al linguaggio. Nel presente studio, confrontiamo il modello con la pointwise mutual information (PMI), una misura di associazione locale tra parole basata sulla loro cooccorrenza. In particolare, abbiamo testato i due indici su un set di dati di priming semantico, mostrando come la PMI riesca a spiegare meglio i dati comportamentali. Alla luce di tali risultati, ciò che è stato tradizionalmente considerato come effetto semantico potrebbe basarsi principalmente su associazioni locali di co-occorrenza lessicale.*

1 Introduction

Over the past two decades, computational semantics has made a lot of progress in the strive for developing techniques that are able to provide human-like estimates of the semantic relatedness between lexical items. Distributional Semantic Models (DSM; Baroni and Lenci, 2010) assume that it is possible to represent lexical meaning based on statistical analyses of the way words are used in large text corpora. Words are modeled as vectors and populate a high-dimensionsional space where similar words tend to cluster together. Meaning relatedness between two words corresponds to the proximity of their vectors; for example, one can approximate relatedness as the cosine of the angle formed by two word-vectors:

$$\cos\theta = \frac{\vec{a}\cdot\vec{b}}{||a||\cdot||b||}$$

DSMs have been proposed as a psychologically plausible models of semantic memory, with particular emphasis on how meaning representations are achieved and structured (e.g. LSA, Landauer and Dumais, 1997; HAL, Lund and Burgess, 1996). So, they can be pitted against human behavior, in search for psychological validation of this modeling. For example, the model's estimates have been used to make reliable predictions about the processing time associated with the stimuli (Baroni et al., 2014; Mandera et al., 2017).

The technique most commonly used to explore semantic processing is the priming paradigm (McNamara, 2005), according to which the recognition of a given word (the *target*) is easier if preceded by a related word (the *prim*e; e.g., cat–dog). Interestingly, facilitation can be observed both when the prime word is fully visible and when it is kept outside of participants' awareness through visual masking (Forster and Davis, 1984; de Wit and Kinoshita, 2015). In this technique, the prime stimulus is displayed shortly, embedded between a forward and a backward string (Figure 1).

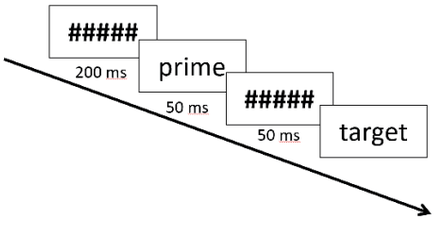


Figure 1: exemplar trial in a masked priming experiment. The prime stimulus is briefly presented (<= 50 ms), between the two masks, before the onset of the target stimulus.

Beside words' distribution, one can be interested in the local association strength between lexical items, starting from the assumption that two words that are often used close to each other, tend to become associated. Yet, a given pair may be often attested only because the two components are in turn highly frequent. Therefore, raw frequency counts are often transformed into some kinds of association measure which can determine if the pair is attested above chance (Evert, 2008). A common method is to compute pointwise mutual information (PMI) between two words, according to the formula:

$$\mathrm{PMI}(w_1,w_2) = \log_2 \frac{p(w_1,w_2)}{p(w_1)p(w_2)}$$

where $p(w_1,w_2)$ corresponds to the probability of the word pair, while $p(w_1)$ and $p(w_2)$ to the individual probabilities of the two components (Church and Hanks, 1990).

PMI has been used to model a wide range of psycholinguistics phenomena, from similarity judgements (Recchia and Jones, 2009) to reading speed (Ellis and Simpson-Vlach, 2009). Moreover, PMI has also been shown to successfully generalize to non-linguistic fields as epistemology and psychology of reasoning (Tentori et al., 2014). On the other hand, PMI has the limit of over-estimating the importance of rare items (Manning and Schütze, 1999).

Despite many DSMs use measures of local association between words like PMI to build contingency matrices, the information conveyed by two similar word-vectors is different from the information conveyed by two highly recurrent words. Cosine similarity is based on "higher order" co-occurrences: two words are similar in the way they are used together with all the other words in the vocabulary. Local measures as PMI instead rely only on the effective co-presence of two given words. Two synonyms like the words *car* and *automobile* are not likely to often appear close to each other in a given text, still they represent the same referent, and therefore expected to be used in similar contexts.

Based on these considerations, PMI and DSMs can be pitted against human behavior, in search for psychological validation of this modeling. In particular, we tested how PMI and cosine proximity predicts priming in a set of data encompassing different prime visibility conditions (masked vs unmasked) and prime durations (33, 50, 200, 1200 ms).

2 Our Study

2.1 Material

All the stimuli used in the current study were italian words. 50 words referring to animals and 50 words referring to tools were used as target stimuli. Each word in this list was paired with three words from the same category, resulting in 300 unique prime-target couples which were divided into three rotations. We add to each rotation 100 additional filler trials which will not be included in the analysis step. More precisely, we used abstract word as target stimuli, paired with animals and tool primes different from those presented in the experimental trials. In this way we ensured that the response to the target was not predictable by the presence of the prime.
Relatedness estimates were obtained by looking at the stimuli distribution across the ItWac corpus, a linguistic database of nearly 2 billion words built through web crawling (Baroni et al., 2009). We downloaded the lemmatized and part-of-speech annotated corpus, freely provided by the authors. All characters were set to lowercase, and special characters were removed together with a list of stop-words.

PMI between the word pairs was computed based on frequency counts gained by sliding a 5-words window along ItWac. Cosine proximity between word vectors was obtained training a word2vec model (Mikolov et al., 2013) on the same corpus. Model's parameters were set according to the WEISS model (Marelli, 2017). All words attested at least 100 times were included in the model, which was trained using the continuous-bag-of-word architecture, a 5-word window and 200 dimensions. The parameter k for negative sampling was set to 10, and the subsampling parameter to 10^{-5}.
Correlations between semantic and lexical variables are shown in Table 1.

	Target length	Target frequency	PMI	cosine
Target length	1			
Target frequency	-.211	1		
PMI	.091	-.205	1	
cosine	.147	-.059	.541	1

Table 1: Correlations between lexical and semantic indexes in our stimulus set.

2.2 Methods

Participants: Overall, 246 volunteers were recruited for the current study, and were assigned to the different prime timing conditions. All subjects were native Italian speakers, with normal or corrected-to-normal vision and no history of neurological or learning diseases.

Apparatus: All stimuli were displayed on a 25'' monitor with a refresh rate of 120 Hz, using MatLab Psychtoolbox. The words and the masks were presented in Arial font 32, in white color against a black background.

Procedure: Participants were engaged in a classic YES/NO task, requiring them to classify the stimuli as members of either the animal or the tool category, according to the instructions. YES-response were always provided with the dominant hand.
Each unique prime-target pair was presented only once to each participant. Experimental sessions included a total of 200 trials, which were divided into two blocks. In one block, subjects were asked to press the yes-button if the target word referred to an animal, while in the other block they were asked to press the yes-button if the target word referred to a tool. The order of the two blocks was counterbalanced across subjects. 10 practice and 2 warm-up trials were presented before each block. Participants could take a short break halfway through each block.
Each trial began with a 750 ms fixation-cross (+). Prime duration was varied across experiments: 33, 50, 200 and 1200 ms respectively. In the former two conditions, prime visibility was prevented through forward and backward visual masks. Finally, the target word was left on the screen until a response was provided.

Prime visibility task. In the experiments with the masked primes, participants were not informed about their presence. This was only revealed after the relevant session, when participants were invited to take part into a prime visibility task requiring them to spot the presence of the letter "n" within the masked word. After the first two examples, where prime duration was increased to 150 ms to ensure visibility, 10 practice and 80 experimental trials were displayed. Prime visibility was quantified through a d–prime analysis carried out on each participant (Green and Swets ,1966).

2.3 Results

Response times (RT) were analyzed on accurate, yes-response trials only. RT were inverse transformed to approximate a normal distribution and employed as a dependent variable in linear mixed-effects regression models. This analysis allows us to control for all the covariates that may have affected the performance, such as trial position in the randomized list, rotation, RT and accuracy on the preceding trial, the response required in the preceding trial, frequency and length of the target. All these variables, together with the two semantic indexes (*PMI* and *cosine proximity*), were entered in the model as fixed effects, while participants and items were considered as random intercepts. Model selection was implemented stepwise, progressively removing those variables whose contribution to goodness of fit was not significant.

In the masked priming data, neither PMI nor cosine proximity were reliable predictors by themselves (p=.298 and p=.206, respectively). However, both indexes interacted with prime visibility as tracked by participants' d–prime ($F_{pmi*d'}$ (1, 9750)= 13.74, p<.001; $F_{cos*d'}$ (1,

9745)= 13.24, p<.001.). As illustrated in Figure 1, the more each participant could see the prime word, the higher the priming effect she displayed.

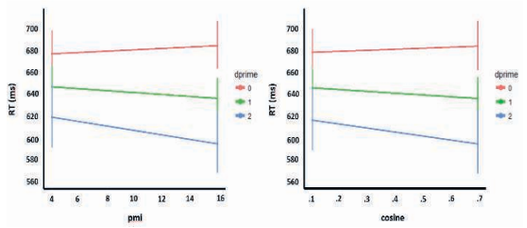


Figure 1. Interaction between d' and prime–target association. Both PMI (left) and cosine proximity (right) effects become stronger as prime visibility (d') increases. Error bars refer to 95% C.I.

In the overt priming data, both PMI and cosine proximity yield a significant main effect (50ms presentation time: F_{pmi}(1,9769)= 10.36, p= .001; F_{cos}(1, 9769)= 8.602, p= .0058), but only PMI significantly predicts priming when both indexes are entered into the model (F_{pmi}(1,9769)= 10.36, p= .001; F_{cos} (1,9769)=0.60, p=.489). Results were very consistent across conditions and showed the same pattern when prime presentation time was 200ms or 1200ms (see Figure 2).

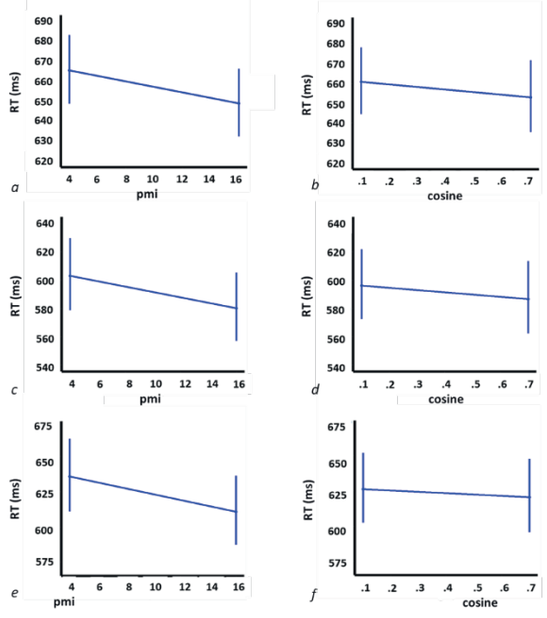


Figure 2. Significant effect of PMI (right) and non-significant effect of cosine proximity (right) across prime presentation times (50ms, 200ms, 1200ms on the first, second and third row respectively). Error bars refer to 95% C.I.

Conclusion

Thanks to the help of computational methods, we provided new insights on the nature of the processing that supports semantic priming. Overall, effects seem to be primarily driven by local word associations as tracked by Pointwise Mutual Information—when semantic priming emerged, PMI effects were consistently stronger and more solid than those related to DSM estimates. This would be in line with previous literature suggesting that the behavior of the human cognitive system may be effectively described by Information Theory principles. For example, Paperno and colleagues (Paperno et al., 2014) showed that PMI is a significant predictor of human judgements of word co–occurrence.

The results from masked priming offer another important insight—some kind of prime visibility may be required for semantic/associative priming to emerge. Other studies have shown genuine semantic effects with subliminally presented stimuli (Bottini et al., 2016). However, they typically used words from small/closed classes (e.g., spatial words, planet names). Conversely, we drew stimuli across the lexicon, and sampled form very large category such as animals and tools; this may point to an effect of target predictability. In general, our data cast some doubts on a wide–across–the–lexicon processing of semantic information outside of awareness.

References

Baroni M., S. Bernardini, A. Ferraresi and E. Zanchetta. (2009). The WaCky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora. *Language Resources and Evaluation* 43 (3): 209-226.

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Vol. 1, pp. 238-247)

Bottini, R., Bucur, M., and Crepaldi, D. (2016). The nature of semantic priming by subliminal spatial words: Embodied or disembodied?. *Journal of Experimental Psychology: General*, *145*(9), 1160.

de Wit, B., and Kinoshita, S. (2015). The masked semantic priming effect is task dependent: Reconsidering the automatic spreading activation process. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 41(4), 1062.

Ellis, N. C., and Simpson-Vlach, R. (2009). Formulaic language in native speakers: Triangulating psy-

cholinguistics, corpus linguistics, and education. *Corpus Linguistics and Linguistic Theory*, *5*(1), 61-78.

Evert, S. (2008). Corpora and collocations. *Corpus linguistics. An international handbook*, 2, 223-233.

Forster, K. I., and Davis, C. (1984). Repetition priming and frequency attenuation in lexical access. *Journal of experimental psychology: Learning, Memory, and Cognition*, *10*(4), 680.

Green D.M. and Swets J.A. (1966). *Signal detection theory and psychophysics.* Wiley New York.

Landauer, T. K., and Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2), 211.

Lund, K., and Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior research methods, instruments, and computers*, 28(2), 203-208.

Mandera, P., Keuleers, E., and Brysbaert, M. (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.

Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT press.

Marelli, M. (2017). Word-Embeddings Italian Semantic Spaces: A semantic model for psycholinguistic research. *Psihologija*, *50*(4), 503-520.

McNamara, T. P. (2005). *Semantic priming: Perspectives from memory and word recognition*. Psychology Press.

Mikolov, P., Chen, K., Corrado, G. S. Dean, J. (2013). Efficient estimation of word representations in vector space. Available from ArXiv:1301.3781.

Paperno, D., Marelli, M., Tentori, K., and Baroni, M. (2014). Corpus-based estimates of word association predict biases in judgment of word co-occurrence likelihood. *Cognitive psychology*, *74*, 66-83.

Recchia, G., and Jones, M. N. (2009). More data trumps smarter algorithms: Comparing pointwise mutual information with latent semantic analysis. *Behavior research methods*, *41*(3), 647-656.

Rohaut, B. and Naccache, L. (2018), What are the boundaries of unconscious semantic cognition? *Eur J Neurosci. . doi:10.1111/ejn.13930.*

Tentori, K., Chater, N., and Crupi, V. (2016). Judging the Probability of Hypotheses Versus the Impact of Evidence: Which Form of Inductive Inference Is More Accurate and Time□Consistent? *Cognitive science*, *40*(3), 758-778.

Online Neural Automatic Post-editing for Neural Machine Translation

Matteo Negri[1] **Marco Turchi**[1] **Nicola Bertoldi**[1,2] **Marcello Federico**[1,2]

[1] Fondazione Bruno Kessler - Trento, Italia
[2] MMT Srl - Trento, Italia
[negri,turchi,bertoldi,federico]@fbk.eu

Abstract

English. Machine learning from user corrections is key to the industrial deployment of machine translation (MT). We introduce the first on-line approach to automatic post-editing (APE), i.e. the task of automatically correcting MT errors. We present experimental results of APE on English-Italian MT by simulating human post-edits with human reference translations, and by applying online APE on MT outputs of increasing quality. By evaluating APE on generic vs. specialised and static vs. adaptive neural MT, we address the question: At what cost on the MT side will APE become useless?

Italiano. *L'apprendimento automatico dalle correzioni degli utenti è fondamentale per lo sviluppo industriale della traduzione automatica (MT). In questo lavoro, introduciamo il primo approccio on-line al post-editing automatico (APE), ovvero il compito di correggere automaticamente gli errori della MT. Presentiamo risultati di online APE su MT da inglese a italiano simulando le correzioni umane con traduzioni manuali già disponibili e utilizzando MT di qualità crescente. Valutando l'APE su MT neurale generica oppure specializzata, statica o adattiva, affrontiamo la domanda di fondo: a fronte di quale costo sul lato MT l'APE diventerà inutile?*

1 Introduction

Automatic Post-editing for MT is a supervised learning task aimed to correct errors in a machine-translated text (Knight and Chander, 1994; Simard et al., 2007). Cast as a problem of "monolingual translation" (from raw MT output into improved text in the same target language), APE has followed a similar evolution to that of MT. As in MT, APE research received a strong boost from shared evaluation exercises like those organized within the well-established WMT Conference on Machine Translation (Chatterjee et al., 2018). In terms of approaches, early MT-like phrase-based solutions (Béchara et al., 2011; Rosa et al., 2013; Lagarda et al., 2015; Chatterjee et al., 2015) have been recently outperformed and replaced by neural architectures that now represent the state of the art (Junczys-Dowmunt and Grundkiewicz, 2016; Chatterjee et al., 2017a; Tebbifakhr et al., 2018; Junczys-Dowmunt and Grundkiewicz, 2018). From the industry standpoint, APE has started to attract MT market players interested in combining the two technologies to support human translation in professional workflows (Crego et al., 2016).

Focusing on this industry-oriented perspective, this paper makes a step further on APE research by exploring an online neural approach to the task. The goal is to leverage human feedback (post edits) to improve on-the-fly a neural APE model without the need of stopping it for fine-tuning or re-training from scratch. Online learning capabilities are crucial (both for APE and MT) in computer-assisted translation scenarios where professional translators operate on suggestions provided by machines. In such scenarios, human corrections represent an invaluable source of knowledge that systems should exploit to enhance users' experience and increase their productivity.

Towards these objectives we provide two contributions. One is the first online approach to neural APE. Indeed, while MT-like online learning techniques have been proposed for phrase-based APE (Ortiz-Martínez and Casacuberta, 2014; Simard and Foster, 2013; Chatterjee et al., 2017b), nothing

has been done yet under the state-of-the-art neural paradigm. In doing this, the other contribution is the first evaluation of neural APE run on the output of neural MT (NMT). So far, published results report significant gains[1] when APE is run to correct the output of a phrase-based MT system. To our knowledge, the true potential of APE with higher quality NMT output has not been investigated yet. The last observation introduces a more general discussion on the relation between MT and APE. Since, by definition, APE's reason of being is the sub-optimal quality of MT output, one might wonder if the level of current MT technology still justifies efforts on APE. Along this direction, our third contribution is an analysis of online neural APE applied to the output of NMT systems featuring different levels of performance. Our competitors range from a generic model trained on large parallel data (mimicking the typical scenario in which industry users – *e.g.* Language Service Providers – rely on web-based services or other black-box systems) to highly customized online models (like those that LSPs would desire but typically cannot afford). Our experiments in this range of conditions aim to shed light on the future of APE from the industry standpoint by answering the question: At what cost on the MT side will APE become useless?

2 Online neural APE

APE training data usually consist of (*src*, *mt*, *hpe*) triplets whose elements are: a source sentence (*src*), its translation (*mt*) and a human correction of the translated sentence (*hpe*). Models trained on such triplets are then used to correct the *mt* element of (*src*, *mt*) test data. Neural approaches to the task have shown their effectiveness in batch conditions, in which a static pre-trained model is run on the whole test corpus. When moving to an online setting, instead, APE systems should ideally be able to continuously evolve by stepwise learning from the interaction with the user. This means that, each time a new post-edit becomes available, the model has to update its parameters on-the-fly in order to produce better output for the next incoming sentence. To this aim, we extend a batch APE model by adding the capability to continuously learn from human corrections of its own output. This is done in two steps:
(1) *Before* post-editing, by means of an instance selection mechanism that updates the model by learning from previously collected triplets that are similar to the input test item (see lines 2-5 in Algorithm 1);
(2) *After* post-editing, by means of a model adaptation procedure that learns from human revisions of the last automatic correction generated by the system (lines 8-10).

Similar to the methods proposed in (Chatterjee et al., 2017b) and (Farajian et al., 2017), the instance-selection technique (first update step) consists of two components: *i)* a knowledge base (*KB*) that is continuously fed with the processed triplets, and *ii)* an information retrieval engine that, given the (*src*, *mt*) test item, selects the most similar triplet (lines 2-3). The engine is simultaneously queried using both *src* and *mt* segments and it returns the triplet that has the highest cosine similarity with both (*Top(R)*). If the similarity is above a threshold τ, a few training iterations are run to update the model parameters (line 5). Depending on the application scenario, *KB* can be pre-filled with the APE training data or left empty and filled only with the incoming triplets. In our experiments, the repository is initially empty.

Algorithm 1: Online neural APE
Require *M*: Trained APE model
Require *Ts*: Stream of test data
Require *KB*: Pool of (*src*, *mt*, *hpe*) triplets
1: **while** pop (*src*, *mt*) from *Ts* **do**
2: $R \leftarrow$ Retrieve ((*src*, *mt*), *KB*)
3: (src_{top}, mt_{top}, hpe_{top}) $\leftarrow$ Top (*R*)
4: **if** Sim ((src_{top}, mt_{top}, hpe_{top}), (*src*, *mt*)) $> \tau$ **do**
5: $M^* \leftarrow$ Update (*M*,(src_{top}, mt_{top}, hpe_{top}))
6: *ape* $\leftarrow$ APE (M^*,(*src*, *mt*))
7: *hpe* $\leftarrow$ HumanPostEdit ((*src*, *ape*))
8: *KB* $\leftarrow$ *KB* $\cup$ (*src*,*mt*,*hpe*)
9: $M^{**} \leftarrow$ Update (M^*,(*src*, *mt*, *hpe*))
10: $M \leftarrow M^{**}$
11: **end while**

Once the *hpe* has been generated, the second update step takes place (line 9) by running few training iterations on the (*src*, *hpe*) pair. When training using only one single data point, the learning rate and the number of epochs have a crucial role because too high/small values can make the training unstable/inefficient. To avoid such problems, we connect the two parameters by applying a time-based decay learning rate that reduces the learning rate when increasing of the number of epochs (*i.e.* $lr = lr/(1+num\_epoch)$). In our experiments, this strategy results in better performance than setting a fixed learning rate.

[1] Up to 7.6 BLEU points at WMT 2017 (Bojar et al., 2017)

3 Experiments

We run our experiments on English-Italian data, by comparing the performance of different neural APE models (batch and online) used to correct the output of NMT systems of increasing quality.

3.1 Data

To train our NMT models we use both generic and in-domain data. Generic data cover a variety of domains. They comprise about 53M parallel sentences collected from publicly-available collections (*i.e.* all the English-Italian parallel corpora available on OPUS[2]) and about 50M sentence pairs from proprietary translation memories. Generic data, whose size is *per se* sufficient to train a competitive general-purpose engine, are used to build our basic NMT model. On top of it, in-domain (information technology) data are used in different ways to obtain improved, domain-adapted models. In-domain data are selected to emulate the online setting of industrial scenarios where input documents are processed sequentially on a sentence-by-sentence basis. They consist in a proprietary translation project of about 421K segments, which are split in training (416K segments) and test (5,472) keeping the sentence order. Post-edits are simulated using references.

To train the APE models we use the English-Italian section of the eSCAPE corpus (Negri et al., 2018). It consists of about 6.6M synthetically-created triplets in which the *mt* element is produced with phrase-based and neural MT systems.

3.2 NMT models

Our NMT models feature increasing levels of complexity, so to represent a range of conditions in which a user (say a Language Service Provider) has access to different resources in terms of MT technology and/or data for training and adaptation. Our systems, ranked in terms of complexity with respect to these two dimensions are:

Generic (G). This model is trained on the large (103M) multi-domain parallel corpus. It represents the situation in which our LSP entirely relies on an off-the-shelf, black-box MT engine that cannot be improved via domain adaptation.

Generic Online (GO). This model extends G with the capability to learn from the incoming human post-edits (5,472 test items). Before and after translation, few training iterations adapt it to the domain of the input document. The adaptation steps implement the same strategies of the online APE system (see §2). This setting represents the situation in which our LSP has access to the inner workings of a competitive online NMT system.

Specialized (S). This model is built by fine-tuning (Luong and Manning, 2015) G on the in-domain training data (416K). It reflects the condition in which our LSP has access both to customer's data and to the inner workings of a competitive *batch* NMT engine. The adaptation routine, however, is limited to the standard approach of performing additional training steps on the in-domain data.

Specialized Online (SO). This model is built by combining the functionalities of GO and S. It uses the in-domain training data for fine-tuning and the incoming (*src*, *hpe*) pairs for online adaptation to the target domain. This setting represents the situation in which our LSP has access to: *i)* customer's in-domain data and *ii)* the inner workings of a competitive *online* NMT engine.

All the models are trained with the ModernMT open source software,[3] which is built on top of OpenNMT-py (Klein et al., 2017). It employs an LSTM-based recurrent architecture with attention (Bahdanau et al., 2014) using 2 bi-directional LSTM layers in the encoder, 4 left-to-right LSTM layers in the decoder, and a dot-product attention model (Luong et al., 2015). In our experiments we used an embeddings' size of 1024, LSTMs of size 1024, and a source and target vocabulary of 32K words, jointly trained with the BPE algorithm (Sennrich et al., 2016). The fact that ModernMT already implements the online adaptation method presented in (Farajian et al., 2017) simplified our tests with online neural APE run on the output of competitive NMT systems (GO and SO).

3.3 APE models

We experiment with two neural APE systems:

Generic APE. This batch system is trained only on generic data (6.6M triplets from eSCAPE) and is similar to those tested in the APE shared task at WMT. The main difference is that the training data are neither merged with in-domain triplets nor selected based on target domain information.

Online APE. This system is trained on the generic data and continuously learns from human post-edits of the test set as described in §2.

[2] `http://opus.lingfil.uu.se` dump of mid June 2017.

[3] `http://github.com/ModernMT/MMT`.

MT Type	MT	Generic APE	Online APE
Generic (G)	40.3	39.0	**47.1**†
Gen. Online (GO)	45.6	41.9	**48.1**†
Specialized (S)	52.1	45.5	**53.5**†
Spec. Online (SO)	**55.0**	47.4	54.8

Table 1: APE performance on NMT outputs of different quality ("†" denotes statistically significant differences wrt. the MT baseline with $p<0.05$).

The two systems are based on a multi-source attention-based encoder-decoder approach similar to (Chatterjee et al., 2017a). It employs a GRU-based recurrent architecture with attention and uses two independent encoders to process the *src* and *mt* segments. Similar to the NMT systems, it is trained on sub-word units by using BPE, with a vocabulary created by selecting to 50K most frequent sub-words. Word embedding and GRU hidden state sizes are set to 1024. Network parameters are optimized with Adagrad (Duchi et al., 2011) with a learning rate of 0.01. A development set randomly extracted from the training data is used to set the similarity threshold used by the online model for the first update step (τ=0.5) as well as the learning rate (0.01) and the number of epochs (3) of both adaptation steps.

4 Results and discussion

APE results computed on different levels of translation quality are reported in Table 1. Looking at the NMT performance, all the adaptation techniques yield significant improvements over the Generic model (G). The large gain achieved via fine-tuning on in-domain data (S: +11.8 BLEU) is further increased when adding online learning capabilities on top of it to create the most competitive Specialized Online system (SO: +14.7).

As expected, the batch APE model trained on generic data only (that is, without in-domain information) is unable to improve the quality of raw MT output. Moreover, although APE results increase with higher translation quality, also the performance distance from the more competitive NMT systems becomes larger (from -1.3 to -7.6 points respectively for G and SO). These results confirm the WMT findings about the importance of domain customization for batch APE (Bojar et al., 2017), and advocate for online solutions capable to maximize knowledge exploitation at test time by learning from user feedback.

Online APE achieves significant[4] improvements not only over the output of G (+6.8) and its online extension GO (+2.5), but also over the specialized model S (+1.4). The gain over GO is particularly interesting: it shows that even when APE and MT use the same in-domain data for online adaptation, the APE model is more reactive to human feedback. Though trained on much smaller generic corpora (6.6M triplets versus 103M parallel sentences), the possibility to leverage richer information in the form of (*src*, *mt*, *pe*) instances at test time seems to have a positive impact. A deeper exploration of this aspect falls out of the scope of this paper and is left as future work.

Also with online APE, the gains become smaller by increasing the MT quality, reaching a point where the system can only approach the highest MT performance of SO (with a non-significant -0.2 BLEU difference). This confirms that correcting the output of competitive NMT engines is a hard task, even for a dynamic APE system that learns from the interaction with the user. However, besides improving its performance by learning from user feedback acquired at test time (similar to the APE system), SO also relies on previous fine-tuning on a large in-domain corpus (similar to S). To answer our initial question ("*At what cost on the MT side will APE become useless?*") it is worth remarking that leveraging in-domain training/adaptation data is a considerable advantage for MT but it comes at a cost that should not be underestimated. In terms of the data itself, collecting enough parallel sentences for each target domain is a considerable bottleneck that limits the scalability of competitive NMT solutions. In addition to that, the technology requirements (*i.e.* having access to the inner workings of the NMT engine) and the computational costs involved (for fine-tuning the generic model) are constraints that few LSPs are probably able to satisfy.

5 Conclusion

We introduced an online neural APE system, which is trained on generic data and only exploits user feedback to improve its performance, and evaluated it on the output of NMT systems featuring increasing complexity and in-domain data demand. Our results show the effectiveness of current APE technology in the typical setting of

[4]Statistical significance is computed with paired bootstrap resampling (Koehn, 2004).

most LSPs while, in terms of resources (especially in-domain data) and technical expertise needed. We also conclude that developing MT engines that make APE useless is still a prerogative of few.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv preprint arXiv:1409.0473*.

Hanna Béchara, Yanjun Ma, and Josef van Genabith. 2011. Statistical Post-Editing for a Statistical MT System. In *Proceedings of the 13th Machine Translation Summit*, pages 308–315, Xiamen, China, September.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 Conference on Machine Translation (WMT17). In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 169–214, Copenhagen, Denmark, September.

Rajen Chatterjee, Marion Weller, Matteo Negri, and Marco Turchi. 2015. Exploring the Planet of the APEs: a Comparative Study of State-of-the-art Methods for MT Automatic Post-Editing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics)*, pages 156–161, Beijing, China, July.

Rajen Chatterjee, M. Amin Farajian, Matteo Negri, Marco Turchi, Ankit Srivastava, and Santanu Pal. 2017a. Multi-source Neural Automatic Post-Editing: FBK's participation in the WMT 2017 APE shared task. In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 630–638, Copenhagen, Denmark, September.

Rajen Chatterjee, Gebremedhen Gebremelak, Matteo Negri, and Marco Turchi. 2017b. Online Automatic Post-editing for MT in a Multi-Domain Translation Environment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 525–535, Valencia, Spain, April.

Rajen Chatterjee, Matteo Negri, Raphael Rubino, and Marco Turchi. 2018. Findings of the WMT 2018 Shared Task on Automatic Post-Editing. In *Proceedings of the Third Conference on Machine Translation*, Brussels, Belgium, October. Association for Computational Linguistics.

Josep Crego, Jungi Kim, Guillaume Klein, Anabel Rebollo, Kathy Yang, Jean Senellart, Egor Akhanov, Patrice Brunelle, Aurelien Coquard, Yongchao Deng, et al. 2016. SYSTRAN's Pure Neural Machine Translation Systems. *arXiv preprint arXiv:1610.05540*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12:2121–2159, July.

M. Amin Farajian, Marco Turchi, Matteo Negri, and Marcello Federico. 2017. Multi-Domain Neural Machine Translation through Unsupervised Adaptation. In *Proceedings of the Second Conference on Machine Translation*, pages 127–137, Copenhagen, Denmark, September.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2016. Log-linear Combinations of Monolingual and Bilingual Neural Machine Translation Models for Automatic Post-Editing. In *Proceedings of the First Conference on Machine Translation*, pages 751–758, Berlin, Germany, August.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2018. Microsoft and University of Edinburgh at WMT2018: Dual-Source Transformer for Automatic Post-Editing. In *Proceedings of the Third Conference on Machine Translation*, Brussels, Belgium, October.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, July.

Kevin Knight and Ishwar Chander. 1994. Automated Post-Editing of Documents. In *Proceedings of AAAI*, volume 94, pages 779–784.

Philipp Koehn. 2004. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the Empirical Methods on Natural Language Processing*, pages 388–395, Barcelona, Spain, July.

Antonio L. Lagarda, Daniel Ortiz-Martínez, Vicent Alabau, and Francisco Casacuberta. 2015. Translating without In-domain Corpus: Machine Translation Post-Editing with Online Learning Techniques. *Computer Speech & Language*, 32(1):109–134.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford Neural Machine Translation Systems for Spoken Language Domains. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT'15)*, pages 76–79, Da Nang, Vietnam, December.

Minh Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *arXiv preprint arXiv:1508.04025*.

Matteo Negri, Marco Turchi, Rajen Chatterjee, and Nicola Bertoldi. 2018. eSCAPE: a Large-scale Synthetic Corpus for Automatic Post-Editing. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May.

Daniel Ortiz-Martínez and Francisco Casacuberta. 2014. The New THOT Toolkit for Fully-Automatic and Interactive Statistical Machine Translation. In *Proceedings of the 14th Annual Meeting of the European Association for Computational Linguistics*, pages 45–48, Gothenburg, Sweden, April.

Rudolf Rosa, David Marecek, and Ales Tamchyna. 2013. Deepfix: Statistical Post-editing of Statistical Machine Translation Using Deep Syntactic Analysis. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 172–179, Sofia, Bulgaria, August.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August.

Michel Simard and George Foster. 2013. PEPr: Post-edit Propagation Using Phrase-based Statistical Machine Translation. In *Proceedings of the XIV Machine Translation Summit*, pages 191–198, Nice, France, September.

Michel Simard, Cyril Goutte, and Pierre Isabelle. 2007. Statistical Phrase-Based Post-Editing. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 508–515, Rochester, New York, April.

Amirhossein Tebbifakhr, Ruchit Agrawal, Rajen Chatterjee, Matteo Negri, and Marco Turchi. 2018. Multi-source Transformer with Combined Losses for Automatic Post-Editing. In *Proceedings of the Third Conference on Machine Translation*, Brussels, Belgium, October.

EnetCollect in Italy

Lionel Nicolas[1], Verena Lyding[1], Luisa Bentivogli[2], Federico Sangati[3], Johanna Monti[3], Irene Russo[4], Roberto Gretter[5], Daniele Falavigna[5]

[1]Institute for Applied Linguistics, Eurac Research, Bolzano
[2]HLT-MT Unit, Fondazione Bruno Kessler, Trento
[3]Department of Literary, Linguistic and Comparative Studies, University of Naples "L'Orientale", Naples
[4] Institute of Computational Linguistics "Antonio Zampolli", CNR, Pisa
[5] SpeechTek Unit, Fondazione Bruno Kessler, Trento

Abstract

English. In this paper, we present the enetCollect[1] COST Action, a large network project, which aims at initiating a new Research and Innovation (R&I) trend on combining the well-established domain of language learning with recent and successful crowdsourcing approaches. We introduce its objectives, and describe its organization. We then present the Italian network members and detail their research interests within enetCollect. Finally, we report on its progression so far.

Italiano. *In questo articolo presentiamo la COST Action enetCollect, un ampio network il cui scopo è avviare un nuovo filone di Ricerca e Innovazione (R&I) combinando l'ambito consolidato dell'apprendimento delle lingue con i più recenti e riusciti approcci di crowdsourcing. Introduciamo i suoi obiettivi e descriviamo la sua organizzazione. Inoltre, presentiamo i membri italiani ed i loro interessi di ricerca all'interno di enetCollect. Infine, descriviamo lo stato di avanzamento finora raggiunto.*

1 Introduction

In this paper, we present the COST network enetCollect that aims at kick-starting an R&I trend for combining language learning with crowdsourcing techniques in order to unlock a crowdsourcing potential for all languages, consisting in learning and teaching activities. This potential will be used to mass-produce language learning material and language-related datasets, such as NLP resources.

We also present enetCollect's Italian members alongside their NLP-related interests. Indeed, NLP heavily relies on language resources and their availability is crucial for the delivery of reliable NLP solutions. Due to high costs of production, resources are often missing, especially for lesser used languages. As enetCollect researches new approaches to tackle such issues, it is a project of particular interest for the Italian NLP community.

EnetCollect connects to ongoing crowdsourcing research, including *Games With A Purpose* approaches (Chamberlain et al., 2013; Lafourcade et al., 2015) for collecting data through gamified tasks (cf. e.g. JeuxDeMots (Lafourcade, 2007), or ZombiLingo (Guillaume et al., 2016)), collaborative approaches such as *Wisdom-of-the-Crowd* initiatives (e.g. dict.cc[2], Wiktionary[3], and Duolingo (von Ahn, 2013)), or general *Human-based Computation* activities (implemented through platforms like Zooniverse[4], Crowd4u[5], etc.).

This paper aims at fostering the participation of the Italian NLP community while further allowing it to benefit from the research and collaboration opportunities enetCollect offers (e.g. research stay grants) for its remaining 2.5 years of funding. Sections 2 and 3 present enetCollect's ambition, and its organization while Section 4 introduces the Italian members and their research interests. Sections 5 and 6 report on achievements up to now and the current state of affairs.

2 Challenge, Motivation and Objectives

Started in March 2017, enetCollect will pursue, until April 2021, the long-term challenge of fostering language learning in Europe and beyond by taking advantage of the ground-breaking nature of crowdsourcing and the immense and ever-

[1]European Network for Combining Language Learning with Crowdsourcing Techniques, Web: (EnetCollect, 2018)

[2]https://www.dict.cc
[3]https://www.wiktionary.org/
[4]https://www.zooniverse.org/
[5]http://crowd4u.org/en/

growing crowd of language learners and teachers[6] to mass-produce language learning content and, at the same time, language-related data such as NLP resources. The prospect of mass-producing language-related data can vastly impact domains such as NLP, which in turn will impact back on language learning by fostering support from various language-related stakeholders (e.g. see Section 4 for NLP-related crowdsourcing scenarios).

As intensifying migration flows (due to economical and geopolitical reasons) increase the diversification of language learner profiles and the demand for learning material, the launch of such an R&I trend is very timely. Indeed, the effectiveness of the existing material runs the risk of gradually falling behind and the varied combinations of languages taught and target groups can hardly be addressed by small-scale initiatives. EnetCollect timely kick-starts an overarching R&I trend to continuously foster various initiatives. Funding-wise, the timing is also favorable as both the increasing need for learning solutions and the problem-solving nature of crowdsourcing are widely acknowledged.

The creation of a new R&I community is addressed through formal *Research Coordination Objectives* aiming at creating a shared knowledge of the subject, at carrying out prototypical experiments and at disseminating promising results while formal *Capacity-Building Objectives* aim at creating the core R&I community, communication means and new initiatives. In Section 5, we report on progress regarding these objectives.

3 Working Groups and Coordinations

EnetCollect makes a working distinction between *explicit* and *implicit* crowdsourcing approaches: while for *explicit* crowdsourcing the crowd intentionally participates (e.g. Wikipedia), for *implicit* crowdsourcing the crowd is not necessarily aware of its participation (e.g. reCaptcha[7]). EnetCollect is organized along five **working groups (WG)** and three support groups called **coordinations**.

Whereas **WG1** focuses on *explicit* crowdsourcing approaches to create data or learning content (e.g. collaboratively creating lessons), **WG2** focuses on *implicit* crowdsourcing approaches for the same purpose (e.g. generating exercise content from language-related resources and collecting the answers to the exercises to correct and extend the resources used). **WG3** focuses on user-oriented design strategies to attract and retain a crowd (e.g. studying the relevance and attractiveness of learner profiling for vocabulary training). **WG4** focuses on studying the functional demands and the existing solutions related to language learning and crowdsourcing (e.g. technical solutions addressing the scalability need of some methods). Finally, **WG5** focuses on application-oriented questions such as ethical issues, legal regulations, and commercialization opportunities.

The five WGs are different content-wise and can be pursued in a parallel fashion. Nonetheless, they remain interdependent in the overarching objective. For example, the boundary between *explicit* and *implicit* crowdsourcing (WG1 and WG2) is sometimes difficult to draw when the crowd is explicitly involved while their actions are being implicitly crowdsourced[8]. Also, any crowdsourcing approach will fail if there is no crowd to rely on (WG3), no technical solution to support its functional needs (WG4), and no appropriate ethical or legal contexts to implement it (WG5). Alongside the WGs, three **coordination groups** on **Dissemination, Exploitation** and **Outreach** are providing standardized support for WG-transversal tasks.

4 Research Interests of Italian Members

The Italian members are currently among the most numerous and active participants to the Action and its events. In addition, the Action coordination (Chair and Grant Holder) is carried out by two Italian members from Eurac Research (see below). Being all related to NLP, enetCollect's Italian partners have a common interest in combining language learning with implicit crowdsourcing (WG2) so as to extend and correct NLP datasets. All crowdsourcing scenarios described hereafter share the same overarching approach: the NLP partner uses an NLP dataset to generate exercise content and both crowdsources and cross-matches the learners' answers in order to validate/discard the data used to generate the exercise content, just like GWAP players validate/discard data while playing. Deriving expert knowledge from cross-matched learners' answers is a challenge enetCollect aims at addressing. Relying on a crowd of

[6] 21% of the Europeans aged over 14 years (~90 millions people, Eurobarometer report, (European Commission, 2012)

[7] https://www.google.com/recaptcha

[8] E.g. crowdsourcing learner essays and their corrections by teachers to create annotated corpora.

learners is however promising in two ways. First, learners should be mostly confronted with exercise content generated from reliable NLP data so as to not undermine their efforts. Their constantly-evaluated proficiency levels thus provide a reliability score for their answers. Second, as a crowd of learners renews itself over time, the set of crowdsourced answers for each question is potentially infinite and their "inferior" reliability is thus compensated by their "superior" quantity.

The **Institute for Applied Linguistics (IAL)** of **Eurac Research** is particularly concerned with research on the three official languages of South Tyrol (Italian, South Tyrolean German and the minority language Ladin). As regards NLP, Italian is the best covered while South Tyrolean is approximated by adapting solutions for standard German and Ladin has barely any coverage. To improve this situation, the IAL aims at crowdsourcing varied NLP resources for South Tyrolean German and Ladin, starting with wide-coverage Part-of-Speech (POS) lexica. The foreseen crowdsourcing scenario is to use POS lexica to generate exercise content for widely adopted exercises such as the one for grouping words according to their properties (e.g. "select all verbs among these five words") or for identifying words within a grid of random letters (e.g. "select five adjectives in the grid". By crowdsourcing the learners' answers, the IAL aims at gradually improving the lexica while continuously adding new entries. As for the targeted crowd of learners, the IAL will build on its long-standing collaborations with schools (Vettori and Abel, 2017; Abel et al., 2014) and is considering to target the local language certification[9], an obligatory exam for public positions for which no dedicated learning tool is currently available online.

The **Human Language Technology - Machine Translation (HLT-MT) research unit** of **Fondazione Bruno Kessler (FBK)** is concerned with MT technologies supporting both human translators and multilingual applications. The creation of dedicated language resources is thus a core activity. Within enetCollect, HLT-MT aims at enriching existing parallel corpora and at enhancing MT evaluation by crowdsourcing multiple translations of the same sentence (Bentivogli et al., 2018). As such translations paraphrase one another, they are also of interest for monolingual NLP purposes. Following the growing number of studies on the language learning usage of MT (Somers, 2001; Niño, 2008; Case, 2015; Dongyun, 2017), HLT-MT focuses on "post-editing" exercises fostering *correction* and *writing* skills where students are presented with a sentence and several possible translations and are asked to choose the most appropriate one and, if necessary, revise it. Existing parallel corpora and state-of-the-art MT systems trained on them will allow to test the learners' skills and generate new translations. While learning, students will thus be trained, evaluated and will sometimes be allowed to correct MT outputs and extend training corpora. For such a crowdsourcing scenario, advanced L2 learners will be targeted, especially those studying Translation Studies for Italian, English and German at partners of the Universities of Trento and Bologna.

The **PARSEME-IT research group**[10] of the **Department of Literary, Linguistic and Comparative Studies, University of Naples "L'Orientale"** aims at improving linguistic representativeness, precision, robustness and computational efficiency of NLP applications (Monti et al., 2017). It researches MultiWord Expressions (MWEs[11]), as a major NLP bottleneck, and investigates their representation in language resources and their integration in syntactic parsing, translation technology, and language learning. The possibility to enhance mono- and multilingual language resources focusing on MWEs is of particular interest, especially with regards to MWE lexica and corpora annotated with MWEs. Accordingly, a set of different exercises engaging students from different degrees (junior high, high school, and undergraduates) are envisioned. For example, exercises to improve lists of Italian MWEs and their correspondences in different languages that ask learners to identify/validate MWEs in monolingual texts and suggest possible translations or ask learners to identify/validate MWEs and their translations in parallel corpora. The targeted students are BA and MA students of the university L'Orientale, especially those attending the translation classes with a solid curriculum in linguistics and Translation Studies.

The **Institute of Computational Linguistics 'Antonio Zampolli' (CNR-ILC)** carries out research at the international, European, national and

[9]Exam for bilingualism, Web: (BZ Alto Adige, 2018)

[10]https://sites.google.com/view/parseme-it/home

[11]Groups of words composing one lexical unit, such as 'tirare le cuoia' (En. kick the bucket)

regional level since 1967. It participated in several EU initiatives on language resource documentation and recently took the lead of the national CLARIN-IT[12] consortium. Its main areas of competence also include Text Processing, NLP, Knowledge Extraction, and Computational Models of Language Usage. Among ILC's resources, ImagAct[13], a multimodal resource about action verbs, represents a starting point for crowdsourcing experiments, where words denoting actions could be explained through videos sharing a semantic core. Crowdsourcing could be used to build these datasets by asking learners to label actions shown in short videos. As shown with middle school pupils (Coppola et al., 2017), analyzing a video illustrating verbs and associating it with words in multiple languages reinforce metalinguistic reasoning (CARAP, 2012). Such combinations of semantic traits and action verbs can also be used for textual entailment.

The **SpeechTEK research unit** of **Fondazione Bruno Kessler (FBK)** is working on Automatic Speech Recognition (ASR) and addresses computer assisted language learning as an application field. In a first project, it aims to automatically assess children's reading capability at primary school. ASR is used to align a given text with the speech read out by a pupil, to highlight its errors and score it. A second project concerns the use of ASR and classification tools to automatically check the proficiency of Italian students aged between 9 and 16 years, in learning both English and German. Both written texts and spoken utterances have to be evaluated, using reference scores related to some proficiency indicators (e.g. pronunciation, fluency, lexical richness) given by human experts. In the first project, corrections of ASR errors can be crowdsourced and used to build more reliable models for assessing reading capabilities of children. Similarly, in the second project crowdsourcing could help both to transcribe and to score the answers uttered by the students. In both cases, crowdsourcing could allow to adapt ASR models and produce more reliable gold standards.

5 Progression of the Network

In this section, the most relevant achievements[14] related to the overall progression of the network are reported in relation to the formal *Research Coordination* and *Capacity-Building Objectives* outlined earlier in Section 2.[15]

Creating a core community of stakeholders. The already large initial number of 68 individual members for 34 participating countries has increased by 67% to 114 members and by 10% to 38 countries. The people subscribed to enetCollect's mailing list have increased by 149% from 79 to 197. Also, 15 financed research stays, lasting 152 days overall, led to intense cooperations.

Building the theoretical framework. The 30 presentations and 39 posters at network meetings and 15 research stays have contributed to the first building blocks of the foreseen theoretical framework, especially with regards to the state-of-the-art review. So far, 3 meetings and 1 training school were organized (168 participations in total).

Communication and outreach. EnetCollect's intranet and website are online for 9 and 7 months and host already a substantial amount of information. 11 mailing lists targeting subsets of members were created and used. 4 calls for research stays and 5 calls for meeting participation were distributed and drew attention (and members) to enetCollect. Aside from one invited talk, several early activities for publications at conferences of related research communities are ongoing.

Funding new initiatives. Funding applications were supported early on, e.g. through the advertisement of specific opportunities or dedicated internal campaigns (e.g. for Marie Sklodowska-Curie Individual Fellowships). Three applications for mid-sized projects were already submitted in the first year, of which two got positively evaluated, and one got funded by a Swiss agency.

6 Conclusion

We presented enetCollect, outlined its key aspects and introduced both its Italian members and their research interests. By harnessing even a fragment of the crowdsourcing potential existing for all languages taught worldwide, enetCollect could trigger changes of noticeable impact for language learning and language-related R&I fields, such as NLP. The fast uptake and overall progression of enetCollect within its first year indicate its relevance and the potential magnitude of its ambition.

[12] www.clarin-it.it

[13] www.imagact.it

[14] See more information on http://enetcollect.eurac.edu.

[15] We do not report on content-related results as these are too numerous and varied and, more importantly, they are (or will be) the focus of different publications authored by the members having achieved them.

References

Andrea Abel, Aivars Glaznieks, Lionel Nicolas, and Egon Stemle. 2014. Koko: an l1 learner corpus for german. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 2414–2421, Reykjavik, Iceland. European Language Resources Association (ELRA).

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2018. Neural versus phrase-based mt quality: An in-depth analysis on englishgerman and englishfrench. *Computer Speech and Language*, 49:52 – 70.

Provincia autonoma di BZ Alto Adige. 2018. Lésame di bilinguismo. Last accessed: 2018-07-20.

Consiglio d'Europa CARAP. 2012. *Le CARAP: Un Cadre de Rfrence pour les Approches Plurielles des Langues et des Cultures, Comptences et Ressources*. Centre Europen pour les Langues Vivantes, Strasbourg Cedex.

Megan Case. 2015. Machine translation and the disruption of foreign language learning activities. *eLearning Papers*, 45:4 – 16.

Jon Chamberlain, Karën Fort, Udo Kruschwitz, Mathieu Lafourcade, and Massimo Poesio. 2013. Using games to create language resources: Successes and limitations of the approach. In Iryna Gurevych and Jungi Kim, editors, *The People's Web Meets NLP*, Theory and Applications of Natural Language Processing, pages 3–44. Springer Berlin Heidelberg.

Daria Coppola, Raffaella Moretti, Irene Russo, and Fabiana Tranchida. 2017. In quante lingue mangi? tecniche glottodidattiche e language testing in classi plurilingui e ad abilit differenziata. In Francesca Strik Lievers Giovanna Marotta, editor, *Strutture linguistiche e dati empirici in diacronia e sincronia*, Studi Linguistici Pisani, pages 199–231. Pisa University Press.

Sun Dongyun. 2017. Application of post-editing in foreign language teaching: Problems and challenges. *Canadian Social Science*, 13(7):1 – 5.

COST Action EnetCollect. 2018. Enetcollect cost website. Last accessed: 2018-07-20.

Directorate-General for Communication European Commission. 2012. Europeans and their languages. Special eurobarometer 386 report, Survey conducted by TNS Opino & Social, and co-ordinated by the European Commission.

Bruno Guillaume, Karën Fort, and Nicolas Lefebvre. 2016. Crowdsourcing complex language resources: Playing to annotate dependency syntax. In *Proceedings of the International Conference on Computational Linguistics (COLING)*, Osaka, Japan.

M. Lafourcade, N. Le Brun, and A. Joubert. 2015. *Games with a Purpose (GWAPS)*. Wiley-ISTWiley-ISTE, July.

Mathieu Lafourcade. 2007. Making people play for lexical acquisition. In *7th Symposium on Natural Language Processing (SNLP 2007)*, Pattaya, Thailand.

Johanna Monti, Maria Pia di Buono, and Federico Sangati. 2017. Parseme-it corpus an annotated corpus of verbal multiword expressions in italian. In *Fourth Italian Conference on Computational Linguistics-CLiC-it 2017*, pages 228–233. Accademia University Press.

Ana Niño. 2008. Evaluating the use of machine translation post-editing in the foreign language class. *Computer Assisted Language Learning*, 21(1):29 – 49.

Harold Somers. 2001. Three perspectives on mt in the classroom. In *Proceedings of the eighth Machine Translation Summit (MT Summit VIII)*, Santiago de Compostela, Galicia, Spain.

Chiara Vettori and Andrea Abel, editors. 2017. *KOLIPSI II. Gli studenti altoatesini e la seconda lingua: indagine linguistica e psicosociale. / Die Sdtiroler SchlerInnen und die Zweitsprache: eine linguistische und sozialpsychologische Untersuchung.* Eurac Research, Bolzano/Bozen.

Luis von Ahn. 2013. Duolingo: learn a language for free while helping to translate the web. In *Proceedings of the 2013 international conference on Intelligent user interfaces*, pages 1–2. ACM.

Towards SMT-Assisted Error Annotation of Learner Corpora

Nadezda Okinina
Eurac Research
viale Druso 1, Bolzano, Italy
nadezda.okinina@eurac.edu

Lionel Nicolas
Eurac Research
viale Druso 1, Bolzano, Italy
lionel.nicolas@eurac.edu

Abstract

English. We present the results of prototypical experiments conducted with the goal of designing a machine translation (MT) based system that assists the annotators of learner corpora in performing orthographic error annotation. When an annotator marks a span of text as erroneous, the system suggests a correction for the marked error. The presented experiments rely on word-level and character-level Statistical Machine Translation (SMT) systems.

Italian. *Presentiamo i risultati degli esperimenti prototipici condotti con lo scopo di creare un sistema basato sulla traduzione automatica (MT) che assista gli annotatori dei corpora degli apprendenti di lingue durante il processo di annotazione degli errori ortografici. Quando un annotatore segna un segmento di testo come errato il sistema suggerisce una correzione dell'errore segnato. Gli esperimenti presentati utilizzano dei sistemi statistici di traduzione automatica (SMT) al livello di parole e di caratteri.*

1 Introduction

Manual error annotation of learner corpora is a time-consuming process which is often a bottleneck in learner corpora research. *"Computer learner corpora are electronic collections of authentic FL/SL textual data assembled according to explicit design criteria for a particular SLA/FLT*[1] *purpose. They are encoded in a standardised and homogeneous way and documented as to their origin and provenance"* (Granger, 2002). Error-annotated learner corpora serve the needs of language acquisition studies and pedagogy development as well as help the creation of natural language processing tools such as automatic language proficiency level checking systems (Hasan et al., 2008) or automatic error detection and correction systems (see Section 2). In this paper we present our first attempts at creating a system that would assist annotators in performing orthographic error annotation by suggesting a correction for specific spans of text selected and marked as erroneous by the annotators. In the prototypical experiments, the suggestions are generated by word-level and character-level SMT systems.

This paper is organized as follows: we review existing approaches to automatic error correction (Section 2), introduce our experiments (Section 3), present the data we used (Section 4), describe and discuss the performed experiments (Section 5) and conclude the paper (Section 6).

2 Related Work

Orthographic errors are mistakes in spelling, hyphenation, capitalisation and word-breaks (Abel et al., 2016). Automatic orthographic error correction can benefit from methods recently developed for grammatical error correction (GEC) such as methods relying on SMT and Neural Machine Translation (NMT) (Chollampatt et al., 2017, Ji et al., 2017, Junczys-Dowmunt et al., 2016, Napoles et al., 2017, Sakaguchi et al., 2017, Schmaltz et al., 2017, Yuan et al., 2016 etc.). These approaches treat error correction as a MT task from incorrect to correct language. In the case of orthographic error correction these "languages" are extremely close, which greatly facilitates the MT task. In that aspect, error correction is similar to the task of translating closely-related languages such as, for example, Mace-

[1] FL: foreign language, SL: second language, SLA: second language acquisition, FLT: foreign language teaching

donian and Bulgarian (Nakov et al., 2012). In our experiments, we rely on the implementation of SMT models provided by the Moses toolkit (Koehn et al., 2007).

SMT and NMT can be easily adapted to new languages, but their performance depends on the amount and quality of the training data. In order to make up for lack of parallel corpora of texts containing language errors and their correct equivalents, various techniques for resource construction have been suggested, such as using the World Wide Web as a corpus (Whitelaw et al., 2009), parsing corrective Wikipedia edits (Grundkiewicz et al., 2014) or injecting errors in error-free text (Ehsan et al., 2013). For our prototypical experiments, we deliberately limit ourselves to the manually-curated high-quality data at our disposal and use existing German error-annotated corpora as training data.

In recent years learner corpora of German have been used for the creation of systems for automatic German children's spelling errors correction (Stüker et al., 2011, Laarmann-Quante, 2017), but no work has been done on automatic orthographic error correction of adult learner texts.

3 Objectives of the Experiments

The particularity of our work is that we focus on a specific use-case where annotators are assisted in error-tagging newly created learner corpora. To ensure the relevance of our system and limit false positives that would hinder its adoption, the targeted use-case is to only suggest corrections while leaving the task of selecting the error to the linguist. Aforementioned GEC systems take as input text containing language errors and produce corrected text. Thus, they may introduce changes in any part of the text, even where no errors are observed. In order to prevent such behavior, we only submit to our system spans of text marked as erroneous by annotators, while leaving out spans of text not containing errors. Therefore, our system is not directly comparable to existing GEC systems.

A given language error may have more than one possible correction, but in the presented research we limit ourselves to orthographic errors that in most cases have only one correction (Nerius et al., 2007). Our system is meant to be used for the creation of new learner corpora in the Institute for Applied Linguistics where learner corpora of German, Italian and English are created and studied (Abel et al., 2013, Abel et al., 2015, Abel et al., 2016, Abel et al., 2017, Zanasi et al., 2018).

Preliminary experiments with the freely available vocabulary-based spell checking tool Hunspell[2] yielded unsatisfactory results (see Section 5.1) and incited us to try SMT in order to train an error-correction system and tune it to the specific nature of our data. We thus performed a series of experiments to perform a preliminary evaluation of the range of performances of different n-gram models when trained on small-scale data (Section 5.1), studied the impact of the similarity between training data and test data to understand which datasets are the most optimal to train our models on (Sections 5.2 and 5.3) and finally made preliminary attempts to improve the performance by optimising the usage of the SMT systems (Section 5.4).

As our systems are not directly comparable to GEC systems, the usual metrics used to evaluate GEC systems are not fully adequate, because they target a similar but different use case. We thus evaluate our systems according to their accuracy that we define as a ratio between the number of suggestions matching the target hypothesis present in the test data (TH)[3] and the whole number of annotated errors. However, accuracy is not the only criteria as it is also important not to disturb the annotators with irrelevant suggestions: it is better not to suggest any TH than to suggest a wrong one. In order to control the ratio between right and wrong suggestions, we also evaluate our systems according to their precision. We define precision as a ratio between the number of suggestions matching the TH and the whole number of suggestions, correct and incorrect, thus excluding the errors for which the system was consulted, but no correction was suggested. Precision is mainly used as a quality threshold which should remain high, whereas our main performance measure is accuracy.

4 Corpora Used

Our experiments rely on three error-annotated learner corpora: KoKo, Falko and MERLIN.

KoKo is a corpus of 1.503 argumentative essays (811.330 tokens) of written German L1[4] from high school pupils, 83% of which are native speakers of German (Abel et al., 2016). It relies

[2] http://hunspell.github.io/

[3] The TH corresponds to a correction associated with each error (Reznicek et al., 2013).

[4] first language, native language

on a very precise error annotation scheme with 29 types of orthographic errors.

The Falko corpus consists of six subcorpora (Reznicek et al., 2012) out of which we are using the subcorpus of 107 error-annotated written texts by advanced learners of L2 [5] German (122.791 tokens).

The MERLIN corpus was compiled from standardized, CEFR[6]-related tests of L2 German, Italian and Czech (Boyd et al., 2014). We are using the German part of MERLIN that contains 1033 learner texts (154.335 tokens): a little bit more than 200 texts for each of the covered CEFR levels (A1, A2, B1, B2, and C1).

Due to the differences in content and format, we do not use all three learner corpora in all the experiments. KoKo is our main corpus, because of its larger size, easy to use format and detailed orthographic error annotation. We use it in training, validation and testing of our SMT systems. Falko is smaller and its format does not allow an easy alignment of orthographic errors, we thus only use it in some experiments as part of the training corpus (Sections 5.1 and 5.2). MERLIN was annotated similarly to KoKo, therefore error-correction results obtained for these two corpora are easily comparable. Furthermore, MERLIN is representative of different levels of language mastery. We thus use it for testing some of our systems (Section 5.2).

As the language model for our character-based SMT systems cannot be generated from the limited amount of data provided by learner corpora, for that purpose we used 3.000.000 sentences of a German news subcorpus from the Leipzig Corpora Collection[7].

5 Prototypical Experiments

5.1 Testing Different N-Gram Models

We started by testing SMT word and character-based language models with various numbers of n-grams in order to understand which one could suffer less from data scarcity and thus best suit our data[8] (Table 1). We used Moses default values for all the other parameters. The systems were trained on a parallel corpus composed of learner texts and their corrected versions from Falko and KoKo. In each fold of the 10-fold validation, 1/10 of KoKo is taken out of the training corpus and used as a validation corpus.

Since our objective was to only observe the overall adequateness of the SMT models, we only attempted to optimise the way the SMT models were used at a later stage (see Section 5.4).

These prototypical experiments showed that all the SMT models have a rather high precision and that, for this amount of training data, the SMT model that performed best is the word 5-gram model. It yielded an encouraging result of 39% of accuracy and 89% of precision, which is far better than the 11% of accuracy and 8% of precision originally obtained with Hunspell. However, 39% of accuracy were obtained by training on Falko and 9/10 of KoKo and validating on 1/10 of KoKo, which would be the configuration we would have towards the end of the annotation of a new learner corpus. We thus proceeded with our experiments by testing how the SMT models would perform at an earlier stage.

	word-grams				character-grams		
	1	3	5	10	6	10	15
Prec.	84%	87%	**89%**	84%	83%	86%	87%
Acc.	32%	37%	**39%**	38%	16%	21%	29%

Table 1: 10-fold validation on KoKo of SMT models trained on KoKo and Falko.

5.2 Testing the Models on New Data

At an early stage of the annotation of a new learner corpus, an error-correction system could be trained on an already existing corpus. We thus tried to apply the different models trained on Falko, KoKo and the newspapers to MERLIN. However, none of the 7 models presented in the previous section achieved more than 13% of accuracy and 70% of precision on the whole MERLIN corpus. Despite that, these experiments highlighted an interesting aspect: all the models performed better on MERLIN texts of higher CEFR levels compared to MERLIN texts of lower CEFR levels (Table 2). We suspect this phenomenon to be due to the fact that the level of language mastery of MERLIN texts of higher CEFR levels is closer to the level of language mastery of KoKo and Falko texts. This observation indicates that the training and test data must attest to the same level of language mastery, because mistakes made by beginner language learners tend to differ noticeably from mistakes made by advanced language learners. Therefore,

[5] second language, foreign language

[6] Common European Framework of Reference for Languages

[7] http://hdl.handle.net/11022/0000-0000-2417-E

[8] The computational results presented have been achieved in part using the Vienna Scientific Cluster (VSC).

using existing learner corpora as training data is a difficult task as most of them target different types of learners with different profiles and bias towards specific kinds of errors.

	A1	A2	B1	B2	C1
Prec.	60%	61%	77%	72%	78%
Acc.	15%	9%	12%	14%	17%

Table 2: precision and accuracy of the word 5-gram model trained on KoKo and Falko when tested on MERLIN texts of different CEFR levels.

5.3 Training and Testing on One Corpus

The results of the previous experiments incited us to train an SMT model on a small part of a corpus and test it on a bigger part of the same corpus in order to observe how an SMT model would behave when trained on an already annotated part of a new learner corpus. We thus performed 3-fold validation experiments with a word 5-gram model taking 1/3 of KoKo as training data and 2/3 of KoKo as test data and obtained 30% of accuracy[9]. This result was much better than 13% of accuracy we had obtained by training SMT systems on KoKo and Falko and testing them on MERLIN. We thus decided to pursue our experiments with KoKo as both training and test data.

In order to observe the evolution of the system's performance with the growth of the corpus, we also trained it on 2/3 of KoKo and tested it on 1/3 of KoKo. Augmenting the training corpus size did not change the system's performance (Table 3, line 1). Such results tend to indicate that most of the performance can be obtained at an earlier stage of the annotation process.

5.4 Improving the Performance

After evaluating the impact of the training data on the system's performance, we switched our focus to the optimisation of the way SMT models were used. First of all, we tried to take into account not only the highest-ranked suggestion of Moses, that in many cases was equal to the error text (i.e. no correction was suggested), but also the lower-ranked suggestions in order to find the highest-ranked suggestion that was different from the error text. This change considerably improved the accuracy for both corpus sizes and only slightly deteriorated the precision (Table 3, line 2).

In order to further improve the performance, we decided to combine the word-based and character-based systems. For this first experiment we chose the best-performing of the word-based systems which is the word 5-gram model and the second best performing of the character-based systems which is the character 10-gram model. We chose the character 10-gram model for practical reasons: it is considerably less resource-consuming than the character 15-gram model. By applying both the word 5-gram and the character 10-gram models to the same data and comparing the overlap in their responses, we verified their degree of complementarity. This experiment showed that only in 18% of cases the word-based and character-based models both suggest a correction (corresponding or not to the TH). In 39% of cases only the word-based system suggests a correction and in 5% of cases only the character-based system suggests a correction. It means that by combining the two systems it is possible to improve the overall performance. We calculated the maximum theoretical accuracy[10] of such a combined system and came to a conclusion that it cannot exceed 53% when trained on 1/3 of KoKo and 60% when trained on 2/3 of KoKo (Table 3, line 3).

By simply giving preference to the word-based model before consulting the character-based model, we almost achieved the maximum theoretical accuracy (Table 3, line 4).

However, we realised that by augmenting the training corpus size, we augmented the accuracy, but slightly deteriorated the precision.

By analysing the performance of different modules (word 5-gram highest-ranked suggestions, word 5-gram lower-ranked suggestions, character 10-gram) on different kinds of errors, we could observe that their performance differs according to types of errors. For example, the lower-ranked suggestions of the word-based model introduce a lot of mistakes in the correction of errors where one word was erroneously written as two separate words (e.g. *Sommer fest* instead

[9]We also calculated the BLEU score for this model and obtained 95%. This result shows that the BLEU score is irrelevant for the evaluation of error correction systems such as ours that cannot introduce errors in error-free spans of text.

[10]The maximum theoretical accuracy would be achieved if it was possible to always choose the right system to consult for each precise error (word-based or character-based) and never consult the system that gave a wrong result when the other system gave a correct result. In that case the maximum potential of both systems would be used.

of *Sommerfest*). We tried to prevent such false corrections by not consulting the lower-ranked suggestions of the word-based model for errors containing spaces. By introducing this rule we succeeded in improving the precision at the cost of loosing some accuracy (Table 3, line 5). This experiment showed that add-hoc rules might not be a workable solution and a more sophisticated approach should be considered if we intend to dynamically combine several systems. In order to obtain better results combining two or more word-based and character-based systems, further experiments should be conducted.

		train. 1/3 valid. 2/3	train. 2/3 valid. 1/3
1	word highest-ranked corr.	30% (88%)	30% (88%)
2	word lower-ranked corr.	48% (84%)	55% (83%)
3	max. theoretical accuracy word lower-ranked + character	53% (85%)	60% (84%)
4	word lower-ranked + character	53% (84%)	59% (83%)
5	word lower-ranked +character with rule on spaces	52% (88%)	57% (88%)

Table 3: accuracy and precision (in brackets) of different systems according to training corpus size (3-fold validation on KoKo).

6 Conclusion

Our preliminary experiments brought us to the conclusion that a SMT system trained on a manually annotated part of a learner corpus can be helpful in error-tagging the remaining part of the same learner corpus: it is possible to train a system that would propose the right correction for half of the orthographic errors outlined by the annotators while proposing very few wrong corrections. Such results are satisfactory enough to start integrating the system into the annotation tool we use to create learner corpora (Okinina et al., 2018).

The combination of a word-based and a character-based systems gave promising results, therefore we intend to continue experimenting with multiple combinations of word-based and character-based systems. We are also considering the possibility to rely on other technologies (Bryant, 2018). As in our experiments we only wanted to observe the range of performances we could expect, we trained our models with the default configuration provided with the MOSES toolkit and did not perform any tuning of the parameters. Future efforts will focus on evaluating how relevant the tuning of parameters can be for such a MT task.

The choice of training data for our experiments was dictated by the availability of high-quality resources. In future experiments we would like to enlarge the spectrum of resources considered for our experiments and work with other languages, in particular with Italian and English.

Acknowledgements

We would like to thank the reviewers as well as our colleagues Verena Lyding and Alexander König for their useful feedback and comments.

References

Abel, A., Konecny, C., Autelli, E.: Annotation and error analysis of formulaic sequences in an L2 learner corpus of Italian, *Third International Learner Corpus Research Conference*, 2015, Book of abstracts, pp. 12-15.

Abel, A., Glaznieks, A., Nicolas, L., Stemle, E.: An extended version of the KoKo German L1 Learner corpus, *Proceedings of the Third Italian Conference on Computational Linguistics CliC-it*, Naples, Italy, 2016, pp. 13-18.

Abel, A., Glaznieks, A.: „Ich weiß zwar nicht, was mich noch erwartet, doch ...“ – Der Einsatz von Korpora zur Analyse textspezifischer Konstruktionen des konzessiven Argumentierens bei Schreibnovizen, *Corpora in specialized communication*, vol. 4, Bergamo, 2013, pp. 101-132.

Abel, A., Vettori, C., Wisniewski, K.: KOLIPSI. Gli studenti altoatesini e la seconda lingua: indagine linguistica e psicosociale, vol. 2, Eurac Research, 2017.

Boyd, A., Hana, J., Nicolas, L., Meurers, D., Wisniewski, K., Abel, A., Schöne, K., Štindlová, B., Vettori, C.: The MERLIN corpus: Learner language and the CEFR, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC),* 2014, pp. 1281-1288.

Bryant, C.: Language Model Based Grammatical Error Correction without Annotated Training Data, *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 2018, pp. 247–253.

Chollampatt, S., Ng, H.: Connecting the Dots: Towards Human-Level Grammatical Error Correction, *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, 2017, pp. 327-333.

Granger, S.: A Bird’s Eye View of Learner Corpus Research. In Granger, S., Hung, J., Petch-Tyson, S.

(eds.), Computer Learner Corpora, Second Language Acquisition and Foreign Language Teaching, Amsterdam & Philadelphia: Benjamins, 2002, pp. 3-33.

Ehsan, N., Faili, H.: Grammatical and context-sensitive error correction using a statistical machine translation framework, *Software – Practice and Experience*, 2013, 43, pp. 187-206.

Grundkiewicz, R., Junczys-Dowmunt, M.: The WikEd Error Corpus: A Corpus of Corrective Wikipedia Edits and Its Application to Grammatical Error Correction. In Przepiórkowski, A., Ogrodniczuk, M. (eds.), Advances in Natural Language Processing. NLP 2014. Lecture Notes in Computer Science, vol. 8686. Springer, Cham, 2014, pp. 478-490.

Hasan, M. M., Khaing, H. O.: Learner Corpus and its Application to Automatic Level Checking using Machine Learning Algorithms, *Proceedings of ECTI-CON*, 2008, pp. 25-28.

Ji, J., Wang, Q., Toutanova, K., Gong, Y., Truong, S., Gao, J.: A Nested Attention Neural Hybrid Model for Grammatical Error Correction, ArXiv e-prints, 2017.

Junczys-Dowmunt, M., Grundkiewicz, R.: Phrase based machine translation is state-of-the-art for automatic grammatical error correction, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics*, Austin, Texas, 2016, pp. 1546–1556.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E.: Moses: Open source toolkit for statistical machine translation, *Proceedings of ACL '07*, Prague, Czech Republic, 2007, pp. 177–180.

Laarmann-Quante, R.: Towards a Tool for Automatic Spelling Error Analysis and Feedback Generation for Freely Written German Texts Produced by Primary School Children, *Proceedings of the Seventh ISCA workshop on Speech and Language Technology in Education*, 2017, pp. 36-41.

Nakov, P., Tiedemann, J.: Combining Word-Level and Character-Level Models for Machine Translation Between Closely-Related Languages, *Proceedings of the 50th Annual Meeting of the Association of Computational Linguistics (ACL)*, 2012, pp. 301-305.

Napoles, C., Sakaguchi, K., Tetreault, J.: JFLEG: A Fluency Corpus and Benchmark for Grammatical Error Corrections, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics,* vol. 2, Short Papers. Association for Computational Linguistics, Valencia, Spain, 2017, pp. 229–234.

Nerius, D. et al.: *Deutsche Orthographie*. 4., neu bearbeitete Auflage. Hildesheim/Zürich/New York: Olms Verlag, 2007.

Okinina, N., Nicolas, L., Lyding, V.: Transc&Anno: A Graphical Tool for the Transcription and On-the-Fly Annotation of Handwritten Documents, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC)*, 2018, pp. 701-705.

Reznicek, M., Lüdeling, A., Hirschmann, H.: Competing Target Hypotheses in the Falko Corpus: A Flexible Multi-Layer Corpus Architecture, *Automatic Treatment and Analysis of Learner Corpus Data*, John Benjamins Publishing Company, Amsterdam/Philadelphia, 2013, pp. 101-123.

Reznicek, M., Lüdeling, A., Krummes, C., Schwantuschke, F.: Das Falko-Handbuch Korpusaufbau und Annotationen, Version 2.0, 2012.

Sakaguchi, K., Post, M., Van Durme, B.: Grammatical Error Correction with Neural Reinforcement Learning, *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, Asian Federation of Natural Language Processing, Taipei, Taiwan, pp. 366–372.

Schmaltz, A., Kim, Y., Rush, A., Shieber, S.: Adapting Sequence Models for Sentence Correction, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2807-2813.

Stüker S., Fay, J., Berkling, K.: Towards Context-dependent Phonetic Spelling Error Correction in Children's Freely Composed Text for Diagnostic and Pedagogical Purposes, *Interspeech*, 2011.

Whitelaw, C., Hutchinson, B., Chung, G., Ellis, G.: Using the Web for Language Independent Spellchecking and Autocorrection, *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, Singapore, 2009, pp. 890-899.

Yuan, Z., Briscoe, T.,: Grammatical Error Correction Using Neural Machine Translation, *Proceedings of NAACL-HLT 2016*, 2016, pp. 380-386.

Zanasi, L., Stopfner, M.: Rilevare, osservare, consultare. Metodi e strumenti per l'analisi del plurilinguismo nella scuola secondaria di primo grado. In Coonan, C., Bier, A., Ballarin, E., La didattica delle lingue nel nuovo milennio. Le sfide dell'internazionalizzazione, Edizioni Ca'Foscari, 2018, pp. 135-148.

Towards Personalised Simplification based on L2 Learners' Native Language

Alessio Palmero Aprosio[†], Stefano Menini[†], Sara Tonelli[†]
Luca Ducceschi[‡], Leonardo Herzog[‡]
[†]FBK, [‡]University of Trento
{aprosio,menini,satonelli@fbk.eu}
luca.ducceschi@unitn.it
leonardo.herzog@studenti.unitn.it

Abstract

English. We present an approach to improve the selection of complex words for automatic text simplification, addressing the need of L2 learners to take into account their native language during simplification. In particular, we develop a methodology that automatically identifies 'difficult' terms (i.e. false friends) for L2 learners in order to simplify them. We evaluate not only the quality of the detected false friends but also the impact of this methodology on text simplification compared with a standard frequency-based approach.

Italiano. *In questo contributo presentiamo un approccio per selezionare le parole complesse da semplificare in modo automatico, tenendo conto della lingua madre dell'utente. Nello specifico, la nostra metodologia identifica i termini 'difficili' (falsi amici) per l'utente per proporne la semplificazione. In questo contesto, viene valutata non soltanto la qualità dei falsi amici individuati, ma anche l'impatto che questa semplificazione personalizzata ha rispetto ad approcci standard basati sulla frequenza delle parole.*

1 Introduction

The task of automated text simplification has been investigated within the NLP community for several years with a number of different approaches, from rule-based ones (Siddharthan, 2010; Barlacchi and Tonelli, 2013; Scarton et al., 2017) to supervised (Bingel and Søgaard, 2016; Alva-Manchego et al., 2017) and unsupervised ones (Paetzold and Specia, 2016), including recent studies using deep learning (Zhang and Lapata, 2017; Nisioi et al., 2017). Nevertheless, only recently researchers have started to build simplification systems that can *adapt* to users, based on the observation that the preceived simplicity of a document depends a lot on the user profile, including not only specific disabilities but also language proficiency, age, profession, etc. Therefore in the last few months the first approaches to personalised text simplification have been proposed at major conferences, with the goal of simplifying a document for different language proficiency levels (Scarton and Specia, 2018; Bingel et al., 2018; Lee and Yeung, 2018).

Along this research line, we present in this paper an approach to perform automated lexical simplification for L2 learners, able to adapt to the user mother tongue. To our knowledge, this is the first work taking into account this aspect and presenting a solution that, given an Italian document and the user's mother tongue as input, selects only the words that the user may find difficult given his/her knowledge of another language. Specifically, we detect and simplify automatically the terms that may be misleading for the user because they are *false friends*, while we do not simplify those that have an orthographically and semantically similar translation in the user native language (so-called *cognates*). In multilingual settings, for instance while teaching, learning or translating a foreign language, these two phenomena have proven to be very relevant (Ringbom, 1986), because the lexical similarities between the two languages in contact have proven to create interferences, favouring or hindering the course of learning.

We compare our approach to the selection of words to be simplified with a standard frequency-based one, in which only the terms that are not listed in De Mauro's Dictionary of Basic Italian[1] are simplified, regardless of the user native

[1] https://dizionario.internazionale.it/

language. Our experiments are evaluated on the Italian-French pair, but the approach is generic.

2 Approach description

Given a document D_i to be simplified, and a native language L_1 spoken by the user, our approach consists of the following steps:

1. **Candidate selection**: for each content word[2] w_i in D_i, we automatically generate a list of words $W_1 \subset L_1$ which are orthographically similar to w_i. In this phase, several orthographical similarity metrics are evaluated. We keep the 5 most-similar terms to w_i.

2. **False friend and cognate detection**: for each of the 5 most similar words in W_1, we classify whether it is a false friend of w_i or not.

3. **Simplification choice**: Based on the output of the previous steps, the system marks w_i as difficult to understand for the user if there are corresponding false friends in L_1. Otherwise, w_i is left in its original form. When a word is marked as difficult, a subsequent simplification module (not included in this work) should try to find an alternative form (such as a synonym, or a description) to make the term more understandable to the user.

2.1 Candidate Selection

A number of similarity metrics have been presented in the past to identify candidate cognates and false friends, see for example the evaluation in Inkpen and Frunza (2005). We choose three of them, motivated by the fact that we want to have at least one ngram-based metric (XXDICE) and one non ngram-based (Jaro/Winkler). To that, we add a more standard metric, Normalized Edit Distance (NED). The three metrics are explained below:

- XXDICE (Brew et al., 1996). It takes in consideration the shared number of extended bigrams[3] and their position relative to two strings S_1 and S_2. The formula is:

$$\mathrm{XX}(S_1, S_2) = \frac{\sum_B \frac{2}{1+(\mathrm{pos}(x)-\mathrm{pos}(y))^2}}{\mathrm{xb}(S_1) + \mathrm{xb}(S_2)}$$

  where B is the set of pairs of shared extended bigrams (x, y), x in S_1 and y in S_2. The functions $\mathrm{pos}(x)$ and $\mathrm{xb}(S)$ return the position of extended bigram x and the number of extended bigrams in string S respectively.

- NED, Normalized Edit Distance (Wagner and Fischer, 1974). A regular Edit Distance calculates the orthographic difference between two strings assigning a cost to any minimum number of edit operations (deletion, substitution and insertion, all with cost of 1) needed to make them equal. NED is obtained by dividing the edit cost by the length of the longest string.

- Jaro/Winkler (Winkler, 1990). The Jaro similarity metric for two strings S_1 and S_2 is computed as follows:

$$J(S_1, S_2) = \frac{1}{3} \cdot \left(\frac{m}{|S_1|} + \frac{m}{|S_2|} + \frac{m-T}{m} \right)$$

  where m is the number of characters in common, provided that they occur in the same (not interrupted) sequence, and T is the number of transpositions of character in S_1 to obtain S_2. The Winkler variation of the metric adds a bias if the two strings share a prefix.

$$JW(S_1, S_2) = J(S_1, S_2) + (1 - J(S_1, S_2))lp$$

  where l is the number of characters of the common prefix of the two strings, up to four, and p is a scaling factor, usually set to 0.1.

Each of these three measures has some disadvantages. For example, we found that Jaro/Winkler metric boosts the similarity of words with the same root. On the other hand, applying NED leads to several pairs of words having the same similarity score. As a result, two words that are close according to a metric can be far using another metric. To overcome this limitation, we balance the three metrics by computing a weighted average of the three scores tuned on a training set. For details, see Section 3.

nuovovocabolariodibase

[2] Content words are words that have a meaning such as names, adjectives, verbs and adverbs. To extract this information, we use the POS tagger included in the Tint pipeline (Aprosio and Moretti, 2018).

[3] An extended bigram is an ordered letter pair formed by deleting the middle letter from any three letter substring of the word.

2.2 False Friend and Cognate Detection

As for false friend and cognate detection, we rely on a SVM-based classifier and train it on a single feature obtained from a multilingual embedding space (Mikolov et al., 2013), where the user language L_1 and the language of the document to be simplified L_2 are aligned. In particular, the feature is the cosine distance between the embeddings of a given content word w_i in the language L_2 and the embedding of its candidate false friends or cognates in L_1. The intuition behind this approach is that two cognates have a shared semantics and therefore a high cosine similarity, as opposed to false friends, whose meanings are generally unrelated. While past approaches to false friend and cognate detection have already exploited monolingual word embeddings (St Arnaud et al., 2017), we employ for our experiments a multilingual setting, so that the semantic distance between the candidate pairs can be measured in their original language without a preliminary translation.

3 Experimental Setup

In our experiments, we consider a setting in which French speakers would like to make Italian documents easier for them to read. Nevertheless, the approach can be applied to any language pair, given that it requires minimal adaptation.

In order to tune the best similarity metrics combination and to train the SVM classifier, a linguist has manually created an Italian-French gold standard, containing pairs of words marked as either cognates or false friends. These terms were collected from several lists available on the web. Overall, the Ita-Fr dataset contains a training set of 1,531 pairs (940 cognates and 591 false friends) and a test set of 108 pairs (51 cognates and 57 false friends).

For the **candidate selection** step, the goal is to obtain for each term w_i in Italian, the 5 French terms with the highest orthographic similarity. Therefore, given w_i, we compute its similarity with each term in a French online dictionary[4] (New, 2006) using the three scores described in the previous section. The lemmas were normalized for accents and diacritics, in order to avoid poor results of the metrics in cases like *général* and *generale*, where the accented *é* character would be considered different with respect to *e*.[5]

In order to identify the best way to combine the three similarity metrics detailed in Section 2.1., we compute all the possibile combinations of weights on 10 groups of 200 word pairs randomly extracted from the 1,531 pairs in the training set, and then keep the combination that scores the highest average similarity.

In Table 1 we report the percentage of times in which the cognate or false friend of w_i in the training set would appear among the 5 most-similar terms extracted from the French online dictionary according to the three different scores in isolation: XX for XXDICE, JW for Jaro/Winkler and NED for Normalized Edit Distance. We also report the best configuration of the three metrics with the corresponding weight to maximise the presence of a cognate or false friend among the 5 most similar terms. We observe that, while the three metrics in isolation yield a similar result, combining them effectively increases the presence of cognates and false friends among the top candidates. This confirms that the metrics capture three different types of similarity, and that it is recommended to take them all into account when performing candidate selection: an approach where evey metric contributes to detecting false friend / cognate candidates outperforms the single metrics.

XX	JW	NED	% Top 5
1.0	-	-	64.6
-	1.0	-	65.6
-	-	1.0	65.9
0.2	0.4	0.4	77.3

Table 1: Analysis of the candidate selection strategy using different metrics in isolation and in combination.

For **false friends and cognates detection**, we proceed as follows. Given a word w_i in Italian, we identify the 5 most similar words in French using the 0.2-0.4-0.4 score introduced before. In case of ties in the 5th positon, we extend the selection to all the candidates sharing the same similarity value.

Each word pair including w_i and one of the 5 most similar words is then classified as false friend or cognate with a SVM using a radial kernel trained on the 1,531 word pairs in the training set. For the multilingual embeddings used to compute

[4] http://www.lexique.org/

[5] For example, NED between *général* and *generale* returns 0.375 when the two strings are not normalized and 0.125 when they are.

the semantic similarity between the Italian words and their candidates, we use the vectors from Bojanowski et al. (2016)[6] trained on Wikipedia data with fastText (Joulin et al., 2016). We chose these resources since they are available both for Italian and French (and several other languages). For the alignement of the semantic spaces of the two languages we use 22,767 Italian-French word pairs collected from an online dictionary.[7]

4 Evaluation

We perform two types of evaluation. In the first one, the goal is to assess whether the system can correctly identify false friends and cognates in a text. In the second one, we want to check what is the difference between the terms simplified by a system with our approach compared with a standard frequency-based simplification system.

For the first evaluation, we manually create a set of 108 Italian sentences containing one false friend or cognate for French speakers taken from the test set. On each term, we run our algorithm and we consider a term a false friend according to two strategies: *a)* if all 5 most similar words in French are classified as false friends, or *b)* if the majority of them are classified as false friends. Results are reported in Table 2.

	P	R	F1
false friends *(a)*	0.75	0.44	0.55
false friends *(b)*	0.57	0.88	0.69

Table 2: False friends classification using setting *(a)* and *(b)*

The evaluation shows that the two settings lead to two different outcomes. In general terms, the first strategy is more conservative and favours Precision, while the second boosts Recall and F1.

As for the second evaluation, on the same set of sentences, we run our algorithm again, this time trying to classify any content word as being a false friend for French speakers or not. We evaluate this component as being part of a simplification system that simplifies only false friends, and we compare this choice with a more standard approach, in which only 'unusual' or 'unfrequent' terms are simplified. This second choice is taken by comparing each content word with De Mauro's Dictionary of Basic Italian and simplifying only those that are not listed among the 7,000 entries of the basic vocabulary.

This evaluation shows that out of 1,035 content words in the test sentences, our simplification approach based on *a)* would simplify 367 words, and 823 if we adopt the strategy *b)*. Based on De Mauro's dictionary, instead, 240 terms would be simplified. Furthermore, there would be only 76 terms simplified using both strategy *a)* and De Mauro's list, and 154 overlaps for strategy *b)*. This shows that the two approaches are rather complementary and based on different principles. This is evident also looking at the evaluated sentences: while considering frequency lists like De Mauro's, terms such as *accademico* and *speleologo* should be simplified because they are not frequently used in Italian, our approach would not simplify them because they have very similar French translations (*académique* and *spéléologue* respectively), and are not classified as false friends by the system. On the other hand, *vedere* would not be simplified in a standard frequency-based system because it is listed among the 2,000 fundamental words in Italian. However, our approach would identify it as a false friend to be simplified because *vider* in French (transl. *svuotare*) is orthographically very similar to *vedere* but has a completely different meaning.

5 Conclusions

In this work, we have presented an approach supporting personalized simplification in that it enables to adapt the selection of difficult words for lexical simplification to the native language of L2 learners. To our knowledge, this is the first attempt to deal with this kind of adaptation. The approach is relatively easy to apply to new languages provided that they have a similar alphabet, since multilingual embeddings are already available and lists of cognates and false friends, although of limited size, can be easily retrieved online.[8]

The work will be extended along different research directions: first, we will evaluate the approach on other language pairs. Then, we will add a lexical simplification module selecting only the words identified as complex by our approach. For

[6] https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

[7] http://dizionari.corriere.it/

[8] See for example the Wiktionary entries at https://en.wiktionary.org/wiki/Category:False_cognates_and_false_friends

this, we can rely on existing simplification tools (Paetzold and Specia, 2015), which could be tuned to adapt also the simplification choices to the user native language, for example by changing the candidate ranking algorithm. Finally, it would be interesting to involve L2 learners in the evaluation, with the goal to measure the effectiveness of different simplification strategies in a real setting.

Acknowledgments

This work has been supported by the European Commission project SIMPATICO (H2020-EURO-6-2015, grant number 692819). We would like to thank Francesca Fedrizzi for her help in creating the gold standard.

References

Fernando Alva-Manchego, Joachim Bingel, Gustavo Paetzold, Carolina Scarton, and Lucia Specia. 2017. Learning how to simplify from explicit labeling of complex-simplified text pairs. In Greg Kondrak and Taro Watanabe, editors, *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017 - Volume 1: Long Papers*, pages 295–305. Asian Federation of Natural Language Processing.

Alessio Palmero Aprosio and Giovanni Moretti. 2018. Tint 2.0: an all-inclusive suite for nlp in italian. In *Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2018)*, Torino, Italy.

Gianni Barlacchi and Sara Tonelli. 2013. ERNESTA: A Sentence Simplification Tool for Children's Stories in Italian. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing: 14th International Conference, CICLing 2013, Samos, Greece, March 24-30, 2013, Proceedings, Part II*, pages 476–487, Berlin, Heidelberg. Springer Berlin Heidelberg.

Joachim Bingel and Anders Søgaard. 2016. Text simplification as tree labeling. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 337–343. Association for Computational Linguistics.

Joachim Bingel, Gustavo Paetzold, and Anders Søgaard. 2018. Lexi: A tool for adaptive, personalized text simplification. In *Proceedings of COLING*. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *CoRR*, abs/1607.04606.

Chris Brew, David McKelvie, et al. 1996. Word-pair extraction for lexicography. In *Proceedings of the 2nd International Conference on New Methods in Language Processing*, pages 45–55.

Diana Inkpen and Oana Frunza. 2005. Automatic identification of cognates and false friends in french and english. In *Proceedings of RANLP*, pages 251–257, 01.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

John Lee and Chak Yan Yeung. 2018. Personalizing lexical simplification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 224–232. Association for Computational Linguistics.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Boris New. 2006. Lexique 3: Une nouvelle base de données lexicales. In *Actes de la Conférence Traitement Automatique des Langues Naturelles (TALN 2006)*.

Sergiu Nisioi, Sanja Stajner, Simone Paolo Ponzetto, and Liviu P. Dinu. 2017. Exploring neural text simplification models. 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 85–91. Association for Computational Linguistics.

Gustavo Paetzold and Lucia Specia. 2015. Lexenstein: A framework for lexical simplification. In *ACL-IJCNLP 2015 System Demonstrations*, ACL, pages 85–90, Beijing, China.

Gustavo H. Paetzold and Lucia Specia. 2016. Unsupervised lexical simplification for non-native speakers. In Dale Schuurmans and Michael P. Wellman, editors, *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 3761–3767. AAAI Press.

H. Ringbom. 1986. Crosslinguistic influence and the foreign language learning process. In E. Kellerman and Smith Sharwood M., editors, *Crosslinguistic Influence in Second Language Acquisition*. Pergamon Press, New York.

Carolina Scarton and Lucia Specia. 2018. Learning simplifications for specific target audiences. In *ACL (2)*, pages 712–718. Association for Computational Linguistics.

Carolina Scarton, Alessio Palmero Aprosio, Sara Tonelli, Tamara Martín Wanton, and Lucia Specia.

2017. Musst: A multilingual syntactic simplification tool. In *Proceedings of the IJCNLP 2017, System Demonstrations*, pages 25–28. Association for Computational Linguistics.

Advaith Siddharthan. 2010. Complex lexico-syntactic reformulation of sentences using typed dependency representations. In *Proceedings of the 6th International Natural Language Generation Conference (INLG 2010)*, Dublin, Ireland.

Adam St Arnaud, David Beck, and Grzegorz Kondrak. 2017. Identifying cognate sets across dictionaries of related languages. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2519–2528, Copenhagen, Denmark, September. Association for Computational Linguistics.

Robert A Wagner and Michael J Fischer. 1974. The string-to-string correction problem. *Journal of the ACM (JACM)*, 21(1):168–173.

William E Winkler. 1990. String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage.

Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 584–594. Association for Computational Linguistics.

Tint 2.0: an All-inclusive Suite for NLP in Italian

Alessio Palmero Aprosio
Fondazione Bruno Kessler
Trento, Italy
aprosio@fbk.eu

Giovanni Moretti
Fondazione Bruno Kessler
Trento, Italy
moretti@fbk.eu

Abstract

English. In this we paper present Tint 2.0, an open-source, fast and extendable Natural Language Processing suite for Italian based on Stanford CoreNLP. The new release includes some improvements of the existing NLP modules, and a set of new text processing components for fine-grained linguistic analysis that were not available so far, including multi-word expression recognition, affix analysis, readability and classification of complex verb tenses.

Italiano. *In questo articolo presentiamo Tint 2.0, una collezione di moduli open-source veloci e personalizzabili per l'analisi automatica di testi in italiano basata su Stanford CoreNLP. La nuova versione comprende alcune migliorie relative ai moduli standard, e l'integrazione di componenti totalmente nuovi per l'analisi linguistica. Questi includono per esempio il riconoscimento di espressioni polirematiche, l'analisi degli affissi, il calcolo della leggibilità e il riconoscimento dei tempi verbali composti.*

1 Introduction

In recent years, Natural Language Processing (NLP) technologies have become fundamental to deal with complex tasks requiring text analysis, such as Question Answering, Topic Classification, Text Simplification, etc. Both research institutions and companies require accurate and reliable software for free and efficient linguistic analysis, allowing programmers to focus on the core of their business or research. While most of the open-source NLP tools freely available on the web (such as Stanford CoreNLP[1] and OpenNLP[2]) are designed for English and sometimes adapted to other languages, there is a lack of this kind of resources for Italian.

In this paper, we present a novel, extended release of Tint (Palmero Aprosio and Moretti, 2016), a suite of ready-to-use modules for Italian NLP. It is free to use, open source, and can be downloaded and used out-of-the-box (see Section 6). Compared to the previous version, the suite has been enriched with several modules for fine-grained linguistic analysis that were not available for Italian before.

2 Related work

There are plenty of linguistic pipelines available for download. Most of them (such as Stanford CoreNLP and OpenNLP) are language independent and, even if they are not available in Italian out-of-the-box, they could be trained in every existing language. A notable example in this direction is UDpipe (Straka and Straková, 2017), a trainable pipeline which performs most of the common NLP tasks and is available in more than 50 languages, and Freeling (Padró and Stanilovsky, 2012), a C++ library providing language analysis functionalities for a variety of languages. There are also some pipelines for Italian, such as TextPro (Emanuele Pianta and Zanoli, 2008), T2K (Dell'Orletta et al., 2014), and TaNL, but none of them are released as open source (and only TextPro can be downloaded and used for free for research purposes). Other single components are unfortunately available only upon request to the authors, for example the AnIta morphological analyser (Tamburini and Melandri, 2012).

In this respect, Tint represents an exception because not only it includes standard NLP modules, for example Named Entity Recognition and

[1] http://stanfordnlp.github.io/CoreNLP/
[2] https://opennlp.apache.org/

Lemmatization, but it also provides within a single framework additional components that are usually available as separate tools, such as the identification of multi-word expressions, the estimation of text complexity and the detection of text reuse.

Multi-word expression identification is a well studied problem, but most of the tools are available or optimized only for English. One of them, jMWE,[3] is written in Java and provides a parallel project[4] that adds compatibility to CoreNLP (Kulkarni and Finlayson, 2011). The mwetoolkit[5] is written in Python and uses a CRF classifier (Ramisch et al., 2010). The word2phrase module of word2vec attempts to learn phrases in a document of any language (Mikolov et al., 2013), but it is more a statistical tool for phrase extraction than for multi-word detection.

As for the assessment of text complexity, READ-IT (Dell'Orletta et al., 2011) is the only existing tool that gathers readability information for an Italian text. However, while the online demo can be used for free without registration, the tool is not available for offline use.

As for text reuse detection, i.e. when an author quotes (or borrows) another earlier or contemporary author, in the last years it has become easier thanks to new algorithms and high availability of texts (Mullen, 2016; Clough et al., 2002; Mihalcea et al., 2006). However, also in this case, no tools are available for Italian.

3 Tool description

The Tint pipeline is based on Stanford CoreNLP (Manning et al., 2014), an open-source framework written in Java, that provides most of the common Natural Language Processing tasks out-of-the-box in various languages. The framework provides also an easy interface to extend the annotation to new tasks and/or languages. Differently from some similar tools, such as UIMA (Ferrucci and Lally, 2004) and GATE (Cunningham et al., 2002), CoreNLP is easy to use and requires only basic object-oriented programming skills to extend it. In Tint, we adopt this framework to: (i) port the most common NLP tasks to Italian; (ii) make it easily extendable, both for writing new modules and replacing existing ones with more customized ones; and (iii) implement some new annotators as wrappers for external tools, such as entity linking, temporal expression identification, keyword extraction.

4 Modules

In this Section, we present a set of Tint modules, briefly describing those that were already included in the first release (Palmero Aprosio and Moretti, 2016) and focusing with more details on novel, more recent ones. While the old modules perform traditional NLP tasks (i.e. morphological analysis), we have recently integrated components for a more fine-grained linguistic analysis of specific phenomena, such as affixation, the identification of multi-word expressions, anglicisms and euphonic "d". These are the outcome of a larger project involving FBK and the Institute for Educational Research of the Province of Trento (Sprugnoli et al., 2018), aimed at studying with NLP tools the evolution of Italian texts towards the so-called neo-standard Italian (Berruto, 2012).

4.1 Already existing modules

As described in (Palmero Aprosio and Moretti, 2016), the Tint pipeline provides a set of pre-installed modules for basic linguistic annotation: tokenization, part-of-speech (POS) tagging, morphological analysis, lemmatization, named entity recognition and classification (NERC), dependency parsing.

Among the modules, two have been implemented from scratch and do not rely on the components available in Stanford CoreNLP: the tokenizer and the morphological analyser (see below). POS tagging, dependency parsing and NERC are performed using the existing modules in CoreNLP, trained on the Universal Dependencies[6] (UD) dataset in Italian (Bosco et al., 2013), and I-CAB (Magnini et al., 2006) respectively.

Additional modules include wrappers for temporal expression extraction and classification with HeidelTime (Strötgen and Gertz, 2013), keyword extraction with Keyphrase Digger (Moretti et al., 2015), and entity linking using DBpedia Spotlight[7] (Daiber et al., 2013) and The Wiki Machine[8] (Giuliano et al., 2009).

Tokenizer: This module provides text segmentation in tokens and sentences. At first, the text is grossly tokenized. Then, in a second step, tokens that need to be put together are merged us-

[3] http://projects.csail.mit.edu/jmwe/
[4] https://github.com/toliwa/CoreNLP-jMWE
[5] http://mwetoolkit.sourceforge.net/PHITE.php
[6] http://universaldependencies.org/
[7] http://bit.ly/dbpspotlight
[8] http://bit.ly/thewikimachine

ing two customizable lists of Italian non-breaking abbreviations (such as "dott." or "S.p.A.") and regular expressions (for e-mail addresses, web URIs, numbers, dates). This second phase uses (De La Briandais, 1959) to speedup the process.

Morphological Analyser: The morphological analyzer module provides the full list of morphological features for each annotated token. The current version of the module has been trained using the Morph-it lexicon (Zanchetta and Baroni, 2005), but it is possible to extend or retrain it with other Italian datasets. In order to grant fast performance, the model storage has been implemented with the mapDB Java library[9] that provides an excellent variation of Cassandras Sorted String Table. To extend the coverage of the results, especially for the complex forms, such as "porta-cene" or "bi-direzionale", the module tries to decompose the token into prefix-root-infix-suffix and tries to recognise the root form.

See Section 5 for an extensive evaluation of the modules.

4.2 New modules

Affixes annotation: This module provides a token-level annotation about word derivatives, based on *derIvaTario* (Talamo et al., 2016).[10] The resource was built segmenting into derivational cycles about 11,000 derivatives and annotating them with a wide array of features. The module uses this resource in input to segment a token into root and affixes, for example *visione* is analysed as *baseLemma=vedere*, *affix=zione* and *allomorph=ione*.

Classification of verbal tenses: Part-of speech tagger and morphological analyzer released with Tint can identify and classify verbs at token level, but sometimes the modality, form and tense of a verb is the result of a sequence of tokens, as in compound tenses such as participio passato, or passive verb forms. For this reason, we include in Tint a new tense module to provide a more complete annotation of multi-token verbal forms. The module supports also the analysis of discontinuous expressions, like for example *ho sempre mangiato*.

Text reuse: Detecting text reuse is useful when, in a document, we want to measure the overlap with a given corpus. This is needed in a number of applications, for example for plagiarism detection, stylometry, authorship attribution, citation analysis, etc. Tint includes now a component to deal with this task, i.e. identifying parts of an input text that overlap with a given corpus. First of all, each sentence of the corpus is compared with the sentences in the processed text using the FuzzyWuzzy package[11], a Java fuzzy string matching implementation: this allows the system not to miss expressions that are slightly different with respect to the texts in the original corpus. In this phase, only long spans of text can be considered, as the probability of an incorrect match on fuzzy comparison grows as soon as the text length decreases. A second step checks whether the overlap involves the whole sentence and, if not, it analyzes the two texts and identifies the number of overlapping tokens. Finally, the Stanford CoreNLP quote annotator[12] is used to catch text reuse that is in between quotes, ignoring the length limitation of the fuzzy comparison.

Readability: In this module, we compute some metrics that can be useful to assess the readability of a text, partially inspired by Dell'Orletta et al. (2011) and Tonelli et al. (2012). In particular, we include the following indices:

- Number of content words, hyphens (using iText Java Library[13]), sentences having less than a fixed number of words, distribution of tokens based on part-of-speech.
- Type-token ratio (TTR), i.e. the ratio between the number of different lemmas and the number of tokens; high TTR indicates a high degree of lexical variation.
- Lexical density, i.e. the number of content words divided by the total number of words.
- Amount of coordinate and subordinate clauses, along with the ratio between them.
- Depth of the parse tree for each sentence: both average and max depth are calculated on the whole text.
- Gulpease formula (Lucisano and Piemontese, 1988) to measure the readability at document level.

[9] http://www.mapdb.org
[10] http://derivatario.sns.it/
[11] https://github.com/xdrop/fuzzywuzzy
[12] https://stanfordnlp.github.io/CoreNLP/quote.html
[13] https://github.com/itext/itextpdf

- Text difficulty based on word lists from De Mauro's Dictionary of Basic Italian[14].

Multi-word expressions: A specific multi-token annotator has been implemented to recognize more than 13,450 multi-word expressions, the so-called 'poliremathiche' (Voghera, 2004), manually collected from various online resources. The list includes verbal, nominal, adjectival and prepositional expressions (e.g. *lasciar perdere*, *società per azioni*, *nei confronti di*, *mezzo morto*). This annotator can identify also discontinuous multi-words. For example, in the expression *andare a genio* (Italian phrase that means "to like") an adverb can be included, as in *andare troppo a genio*. Similarly, in such phrases one can find nouns and adjectives (e.g. *lasciare Antonio a piedi*, where *lasciare a piedi* is an Italian multiword for *leave stranded*).

Anglicisms: A list of more than 2,500 anglicisms, collected from the web, is included in the last release of Tint, and a particular annotator identifies them in the text and distinguishes between adapted ("chattare", "skillato") and non-adapted anglicisms ("spread", "leadership"). This module can then be used to track the use of borrowings from English in Italian texts, a phenomenon much debated in the media and among scholars (Fanfani, 1996; Furiassi, 2008).

Euphonic "D": For euphonic reasons, the preposition *a*, and the conjunctions *e* and *o* usually become *ad*, *ed*, *od* when the subsequent word begins with *a*, *e*, *o* respectively. While traditionally this rule was applied to every vowel, a more recent grammatical rule has established that the euphonic 'd' should be limited to cases in which it is followed by the same vowel, for example *ed ecco* vs. *e ancora*[15]. Tint provides an annotator that identifies this phenomenon, and classifies each instance as correct, if it follows the aforementioned rule, or incorrect in all the other cases.

Corpus statistics: A collection of CoreNLP annotators have been developed to extract statistics that can be used, for instance, to analyse traits of interest in texts. More specifically, the provided modules can mark and compute words and sentences based on token, lemma, part-of-speech and word position in the sentence.

[14] http://bit.ly/nuovo-demauro

[15] http://bit.ly/crusca-d-eufonica

5 Evaluation

Tint includes a rich set of tools, evaluated separately. In some cases, an evaluation based on the accuracy is not possible, because of the lack of available gold standard or because the tool outcome is not comparable to other tools' ones.

When possible, Tint is compared with existing pipelines that work with the Italian language: Tanl (Attardi et al., 2010), TextPro (Pianta et al., 2008) and TreeTagger (Schmid, 1994).

In calculating speed, we run each experiment 10 times and consider the average execution time. When available, multi-thread capabilities have been disabled. All experiments have been executed on a 2,3 GHz Intel Core i7 with 16 GB of memory.

The Tanl API is not available as a downloadable package, but it's only usable online through a REST API, therefore the speed may be influenced by the network connection.

No evaluation is performed for the Tint annotators that act as wrappers for an external tools (temporal expression tagging, entity linking, keyword extraction).

5.1 Tokenization and sentence splitting

For the task of tokenization and sentence splitting, Tint outperforms in speed both TextPro and Tanl (see Table 1).

System	Speed (tok/sec)
Tint	**80,000**
Tanl API	30,000
TextPro 2.0	35,000

Table 1: Tokenization and sentence splitting speed.

5.2 Part-of-speech tagging

The evaluation of the part-of-speech tagging is performed against the test set included in the UD dataset, containing 10K tokens. As the tagset used is different for different tools, the accuracy is calculated only on five coarse-grained types: nouns (N), verbs (V), adverbs (B), adjectives (A) and other (O). Table 2 shows the results.

5.3 Lemmatization

Like part-of-speech tagging, lemmatization is evaluated, both in terms of accuracy and execu-

[16] The (considerable) speed of TreeTagger includes both lemmatization and part-of-speech tagging.

System	Speed (tok/sec)	Accuracy
Tint	28,000	**98%**
Tanl API	20,000	n.a.
TextPro 2.0	20,000	96%
TreeTagger	**190,000**[16]	92%

Table 2: Evaluation of part-of-speech tagging.

tion time, on the UD test set. When the lemma is guessed starting form a morphological analysis (such as in Tint and TextPro), the speed is calculated by including both tasks. Table 3 shows the results. All the tools reach the same accuracy of 96% (with minor differences that are not statistically significant).

System	Speed (tok/sec)	Accuracy
Tint	97,000	96%
TextPro 2.0	9,000	96%
TreeTagger	**190,000**[16]	96%

Table 3: Evaluation of lemmatization.

5.4 Named Entity Recognition

For Named Entity Recognition, we evaluate and compare our system with the test set available on the I-CAB dataset. We consider three classes: PER, ORG, LOC. In training Tint, we extracted a list of persons, locations and organizations by querying the Airpedia database (Palmero Aprosio et al., 2013) for Wikipedia pages classified as `Person`, `Place` and `Organisation`, respectively. Table 4 shows the results of the named entity recognition task.

System	Speed	P	R	F_1
Tint	**30,000**	**84.37**	79.97	**82.11**
TextPro 2.0	4,000	81.78	**80.78**	81.28
Tanl API	16,000	72.89	52.50	61.04

Table 4: Evaluation of the NER.

5.5 Dependency parsing

The evaluation of the dependency parser is performed against Tanl (Attardi et al., 2013) and TextPro (Lavelli, 2013) w.r.t the usual metrics Labeled Attachment Score (LAS) and Unlabeled Attachment Score (UAS). Table 5 shows the results: the Tint evaluation has been performed on the UD test data; LAS and UAS for TextPro and Tanl is taken directly from the Evalita 2011 proceedings (Magnini et al., 2013).

System	Speed	LAS	UAS
Tint	**9,000**	84.67	87.05
TextPro 2.0	1,300	87.30	91.47
Tanl (DeSR)	900	**89.88**	**93.73**

Table 5: Evaluation of the dependency parsing.

6 Tint distribution

The Tint pipeline is released as an open source software under the GNU General Public License (GPL), version 3. It can be download from the Tint website[17] as a standalone package, or it can be integrated into an existing application as a Maven dependency. The source code is available on Github.[18]

The tool is written using the Stanford CoreNLP paradigm, therefore a third part software can be integrated easily into the pipeline.

7 Conclusions and Future Works

In this paper, we presented the new release of Tint, a simple, fast and accurate NLP pipeline for Italian, based on Stanford CoreNLP. In the new version, we have fixed some bugs and improved some of the existing modules. We have also added a set of components for fine-grained linguistics analysis that were not available so far.

In the future, we plan to improve the suite and extend it with additional modules, also based on the feedback from the users through the github project page. We are currently working on new modules, in particular Word Sense Disambiguation (WSD) based on linguistic resources such as MultiWordNet (Pianta et al., 2002) and Semantic Role Labelling, by porting to Italian resources such as FrameNet (Baker et al., 1998), now available only in English.

The Tint pipeline will also be integrated in PIKES (Corcoglioniti et al., 2016), a tool that extracts knowledge from English texts using NLP and outputs it in a queryable form (such RDF triples), so to extend it to Italian.

Acknowledgments

The research leading to this paper was partially supported by the EU Horizon 2020 Programme via the SIMPATICO Project (H2020-EURO-6-2015, n. 692819).

[17] `http://tint.fbk.eu/`
[18] `https://github.com/dhfbk/tint/`

References

G. Attardi, S. Dei Rossi, and M. Simi. 2010. The Tanl Pipeline. In *Proc. of LREC Workshop on WSPP*.

Giuseppe Attardi, Maria Simi, and Andrea Zanelli. 2013. Tuning desr for dependency parsing of italian. In *Evaluation of Natural Language and Speech Tools for Italian*, pages 37–45. Springer.

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics.

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.

Paul Clough, Robert Gaizauskas, Scott SL Piao, and Yorick Wilks. 2002. Meter: Measuring text reuse. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 152–159. Association for Computational Linguistics.

Francesco Corcoglioniti, Marco Rospocher, and Alessio Palmero Aprosio. 2016. A 2-phase frame-based knowledge extraction framework. In *Proc. of ACM Symposium on Applied Computing (SAC'16)*.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, and Valentin Tablan. 2002. Gate: An architecture for development of robust hlt applications. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 168–175, Stroudsburg, PA, USA. Association for Computational Linguistics.

Joachim Daiber, Max Jakob, Chris Hokamp, and Pablo N. Mendes. 2013. Improving efficiency and accuracy in multilingual entity extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*.

Rene De La Briandais. 1959. File searching using variable length keys. In *Papers Presented at the the March 3-5, 1959, Western Joint Computer Conference*, IRE-AIEE-ACM '59 (Western), pages 295–298, New York, NY, USA. ACM.

Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2011. 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, pages 73–83, Stroudsburg, PA, USA. Association for Computational Linguistics.

Felice Dell'Orletta, Giulia Venturi, Andrea Cimino, and Simonetta Montemagni. 2014. T2kˆ 2: a system for automatically extracting and organizing knowledge from texts. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.

Christian Girardi Emanuele Pianta and Roberto Zanoli. 2008. The textpro tool suite. In Bente Maegaard Joseph Mariani Jan Odijk Stelios Piperidis Daniel Tapias Nicoletta Calzolari (Conference Chair), Khalid Choukri, editor, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Massimo Fanfani. 1996. Sugli-anglicismi nell"italiano contemporaneo (xiv). *Lingua nostra*, 57(2):72–91.

David Ferrucci and Adam Lally. 2004. Uima: An architectural approach to unstructured information processing in the corporate research environment. *Nat. Lang. Eng.*, 10(3-4):327–348, September.

Cristiano Furiassi. 2008. *Non-adapted Anglicisms in Italian: Attitudes, frequency counts, and lexicographic implications*. Cambridge Scholars Publishing.

Claudio Giuliano, Alfio Massimiliano Gliozzo, and Carlo Strapparava. 2009. Kernel methods for minimally supervised wsd. *Comput. Linguist.*, 35(4):513–528, December.

Nidhi Kulkarni and Mark Alan Finlayson. 2011. jmwe: A java toolkit for detecting multi-word expressions. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pages 122–124. Association for Computational Linguistics.

Alberto Lavelli. 2013. An ensemble model for the evalita 2011 dependency parsing task. In *Evaluation of Natural Language and Speech Tools for Italian*, pages 30–36. Springer.

Pietro Lucisano and Maria Emanuela Piemontese. 1988. GULPEASE: una formula per la predizione della difficoltà dei testi in lingua italiana. *Scuola e città*, 3(31):110–124.

Bernardo Magnini, Emanuele Pianta, Christian Girardi, Matteo Negri, Lorenza Romano, Manuela Speranza, Valentina Bartalesi Lenzi, and Rachele Sprugnoli. 2006. I-cab: the italian content annotation bank. In *Proceedings of LREC*, pages 963–968. Citeseer.

Bernardo Magnini, Francesco Cutugno, Mauro Falcone, and Emanuele Pianta. 2013. *Evaluation of Natural Language and Speech Tool for Italian: International Workshop, EVALITA 2011, Rome, January 24-25, 2012, Revised Selected Papers*, volume 7689. Springer.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Rada Mihalcea, Courtney Corley, Carlo Strapparava, et al. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *AAAI*, volume 6, pages 775–780.

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 26*, pages 3111–3119. Curran Associates, Inc.

Giovanni Moretti, Rachele Sprugnoli, and Sara Tonelli. 2015. Digging in the dirt: Extracting keyphrases from texts with kd. *CLiC it*, page 198.

Lincoln Mullen, 2016. *textreuse: Detect Text Reuse and Document Similarity*. R package version 0.1.4.

Lluís Padró and Evgeny Stanilovsky. 2012. Freeling 3.0: Towards wider multilinguality. In *LREC2012*.

A. Palmero Aprosio and G. Moretti. 2016. Italy goes to Stanford: a collection of CoreNLP modules for Italian. *ArXiv e-prints*, September.

Alessio Palmero Aprosio, Claudio Giuliano, and Alberto Lavelli. 2013. Automatic expansion of DBpedia exploiting Wikipedia cross-language information. In *Proceedings of the 10th Extended Semantic Web Conference*.

Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. 2002. Developing an aligned multilingual database. In *Proc. 1st Intl Conference on Global WordNet*. Citeseer.

Emanuele Pianta, Christian Girardi, and Roberto Zanoli. 2008. The textpro tool suite. In *LREC*. Citeseer.

Carlos Ramisch, Aline Villavicencio, and Christian Boitet. 2010. Multiword expressions in the wild?: the mwetoolkit comes in handy. In *Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations*, pages 57–60. Association for Computational Linguistics.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees.

Rachele Sprugnoli, Sara Tonelli, Alessio Palmero Aprosio, and Giovanni Moretti. 2018. Analysing the evolution of students' writing skills and the impact of neo-standard italian with the help of computational linguistics. In *Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2018)*, Torino, Italy.

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. Association for Computational Linguistics.

Jannik Strötgen and Michael Gertz. 2013. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298.

Luigi Talamo, Chiara Celata, and Pier Marco Bertinetto. 2016. DerIvaTario: An annotated lexicon of Italian derivatives. *Word Structure*, 9(1):72–102.

Fabio Tamburini and Matias Melandri. 2012. Anita: a powerful morphological analyser for italian. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uur Doan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Sara Tonelli, Ke Tran Manh, and Emanuele Pianta. 2012. Making readability indices readable. In *Proceedings of the First Workshop on Predicting and Improving Text Readability for target reader populations*, pages 40–48, Montréal, Canada, June. Association for Computational Linguistics.

Miriam Voghera. 2004. Polirematiche. *La formazione delle parole in italiano*, pages 56–69.

Eros Zanchetta and Marco Baroni. 2005. Morph-it! a free corpus-based morphological resource for the italian language. *Corpus Linguistics 2005*, 1(1).

MEDEA: Merging Event knowledge and Distributional vEctor Addition

Ludovica Pannitto
CoLing Lab, University of Pisa
ellepannitto@gmail.com

Alessandro Lenci
CoLing Lab, University of Pisa
alessandro.lenci@unipi.it

Abstract

English. The great majority of compositional models in distributional semantics present methods to compose distributional vectors or tensors in a representation of the sentence. Here we propose to enrich the best performing method (vector addition, which we take as a baseline) with distributional knowledge about events, outperforming our baseline.

Italiano. *La maggior parte dei modelli proposti nell'ambito della semantica disribuzionale composizionale si basa sull'utilizzo dei soli vettori lessicali. Proponiamo di arricchire il miglior modello presente in letteratura (la somma di vettori, che consideriamo come baseline) con informazione distribuzionale sugli eventi elicitati dalla frase, migliorando sistematicamente i risultati della baseline.*

1 Compositional Distributional Semantics: Beyond vector addition

Composing word representations into larger phrases and sentences notoriously represents a big challenge for distributional semantics (Lenci, 2018). Various approaches have been proposed ranging from simple arithmetic operations on word vectors (Mitchell and Lapata, 2008), to algebraic compositional functions on higher-order objects (Baroni et al., 2014; Coecke et al., 2010), as well as neural networks approaches (Socher et al., 2010; Mikolov et al., 2013).

Among all proposed compositional functions, vector addition still shows the best performances on various tasks (Asher et al., 2016; Blacoe and Lapata, 2012; Rimell et al., 2016), beating more complex methods, such as the Lexical Functional Model (Baroni et al., 2014). However, the success of vector addition is quite puzzling from the linguistic and cognitive point of view: the meaning of a complex expression is not simply the sum of the meaning of its parts, and the contribution of a lexical item might be different depending on its syntactic as well as pragmatic context.

The majority of available models in literature assumes the meaning of complex expressions like sentences to be a vector (i.e., an embedding) projected from the vectors representing the content of its lexical parts. However, as pointed out by Erk and Padó (2008), while vectors serve well the cause of capturing the semantic relatedness among lexemes, this might not be the best choice for more complex linguistic expressions, because of the limited and fixed amount of information that can be encoded. Moreover events and situations, expressed through sentences, are by definition inherently complex and structured semantic objects. Actually, assuming the equation "meaning is vector" is eventually too limited even at the lexical level.

Psycholinguistic evidence shows that lexical items activate a great amount of generalized event knowledge (GEK) (Elman, 2011; Hagoort and van Berkum, 2007; Hare et al., 2009), and that this knowledge is crucially exploited during online language processing, constraining the speakers' expectations about upcoming linguistic input (McRae and Matsuki, 2009). GEK is concerned with the idea that the lexicon is not organized as a dictionary, but rather as a network, where words trigger expectations about the upcoming input, influenced by pragmatic knowledge along with lexical knowledge. Therefore sentence comprehension can be phrased as the identification of the event that best explains the linguistic cues used in the input (Kuperberg and Jaeger, 2016).

In this paper, we introduce **MEDEA**, a compositional distributional model of sentence meaning which integrates vector addition with GEK activated by lexical items. MEDEA is directly inspired by the model in Chersoni et al. (2017a) and relies on two major assumptions:

- lexical items are represented with embeddings within a network of syntagmatic relations encoding prototypical knowledge about events;
- the semantic representation of a sentence is a structured object incrementally integrating the semantic information cued by lexical items.

We test MEDEA on two datasets for compositional distributional semantics in which addition has proven to be very hard to beat. At least, before meeting MEDEA.

2 Introducing MEDEA

MEDEA consists of two main components: i.) a **Distributional Event Graph** (DEG) that models a fragment of semantic memory activated by lexical units (Section 2.1); ii.) a **Meaning Composition Function** that dynamically integrates information activated from DEG to build a sentence semantic representation (Section 2.2).

2.1 Distributional Event Graph

We assume a broad notion of *event*, corresponding to any **configuration of entities, actions, properties, and relationships**. Accordingly, an event can be a complex relationship between entities, as the one expressed by the sentence *The student read a book*, but also the association between an individual and a property, as expressed by the noun phrase *heavy book*.

In order to represent the GEK cued by lexical items during sentence comprehension, we explored a graph based implementation of a distributional model, for both theoretical and methodological reasons: in graphs, structural-syntactic information and lexical information can naturally coexist and be related, moreover vectorial distributional models often struggle with the modeling of dynamic phenomena, as it is often difficult to update the recorded information, while graphs are more suitable for situations where relations among items change overtime. The data structure would ideally keep track of each event automatically retrieved from corpora, thus indirectly containing information about schematic or underspecified events, by abstracting over one or more participants from each recorded instance. Events are cued by all the potential participants to the event.

The nodes of DEG are lexical embeddings, and edges link lexical items participating to the same events (i.e., its syntagmatic neighbors). Edges are weighted with respect to the statistical salience of the event given the item. Weights, expressed in terms of a statistical association measure such as *Local Mutual Information*, determine the event activation strength by linguistic cues.

In order to build DEG, we automatically harvested events from corpora, using syntactic relations as an approximation of semantic roles of event participants. From a dependency parsed sentence we identified an event by selecting a semantic head (verb or noun) and grouping all its syntactic dependents together (Figure 1). Since we expect each participant to be able to trigger the event and consequently any of the other participants, a relation can be created and added to the graph from each subset of each group extracted from sentence.

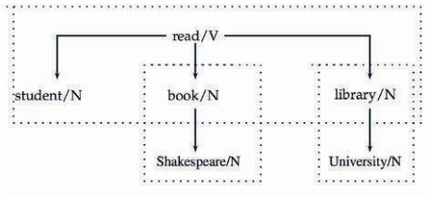


Figure 1: Dependency analysis for the sentence *The student is reading the book about Shakespeare in the university library*. Three events are identified (dotted boxes).

The resulting structure is therefore a weighted hypergraph, as it contains relations holding among groups of nodes, and a labeled multigraph, since each edge or hyperedge is labeled in order to represent the syntactic pattern holding in the group.

As graph nodes are embeddings, given a lexical cue w, DEG can be queried in two modes:

- retrieving the most similar nodes to w (i.e., its paradigmatic neighbors), using a standard vector similarity measure like the cosine (Table 1, top row);
- retrieving the closest associates of w (i.e., its syntagmatic neighbors), using the weights on the graph edges (Table 1, bottom row).

para. neighbors	essay/N, anthology/N, novel/N, author/N, publish/N, biography/N, autobiography/N, nonfiction/N, story/N, novella/N
synt. neighbors	publish/V, write/V, read/V, include/V, child/N, series/N, have/V, buy/V, author/N, contain/V

Table 1: The 10 nearest paradigmatic (top) and syntagmatic (bottom) neighbours of *book/N*, extracted from DEG. By further restricting the query on the graph neighbors, we can obtain for instance typical subjects of *book* as a direct object (*people/N, child/N, student/N, etc.*).

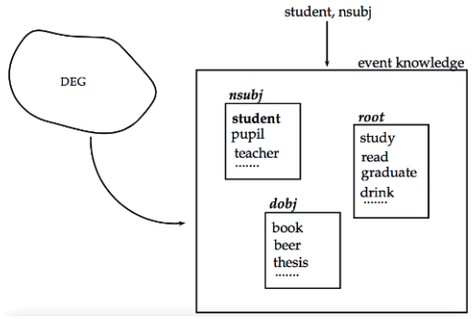


Figure 2: The image shows the internal architecture of a piece of EK retrieved from DEG. The interface with DEG is shown on the left side of the picture, each internal list of *neighbors* is labeled with their expected syntactic role in the sentence. All the items are intended to be embeddings.

2.2 Meaning Composition Function

In MEDEA, we model sentence comprehension as the creation of a semantic representation SR, which includes two different yet interacting information tiers that are equally relevant in the overall representation of sentence meaning: i.) the *lexical meaning* component (LM), which is a context-independent tier of sentence meaning that accumulates the lexical content of the sentence, as traditional models do; ii.) an *active context* (AC), which aims at representing the most probable event, in terms of its participants, that can be reconstructed from DEG portions cued by lexical items. This latter component corresponds to the GEK activated by the single lexemes (or by other contextual elements) and integrated into a semantically coherent structure representing the sentence interpretation. It is incrementally updated during processing, when a new input is integrated into existing information.

2.2.1 Active Context

Each lexical item in the input activates a portion of GEK that is integrated into the current AC through a process of mutual re-weighting that aims at maximizing the overall semantic coherence of the SR.

At the outset, no information is contained in the AC of the sentence. When new *lexeme - syntactic role* pair $\langle w_i, r_i \rangle$ (e.g., *student - nsbj*) are encountered, expectations about the set of upcoming roles in the sentences are generated from DEG (figure 2). These include: i.) expectations about the role filled by the lexeme itself, which consists of its vector (and possibly its *p-neighbours*); ii.) expectations about sentence structure and other participants, which are collected in weighted list of vectors of its *s-neighbours*.

These expectations are then weighted with respect to what is already in the AC, and the AC is similarly adapted to the ewly retrieved information: each weighted list is represented with the weighted centroid of its top elements, and each element of a weighted lists is re-ranked according to its cosine similarity with the correspondent centroid (e.g., the newly retrieved weighted list of *subjects* is ranked according to the cosine similarity of each item in the list with the weighted centroid of *subjects* available in AC).

The final semantic representation of a sentence consists of two vectors, the **lexical meaning vector** ($\overrightarrow{LM}$) and the **event knowledge vector** ($\overrightarrow{AC}$), which is obtained by composing the weighted centroids of each role in AC.

3 Experiments

3.1 Datasets

We wanted to evaluate the contribution of activated event knowledge in a sentence comprehension task. For this reason, among the many existing datasets concerning entailment or paraphrase detection, we chose RELPRON (Rimell et al., 2016), a dataset of subject and object relative clauses, and the transitive sentence similarity dataset presented in Kartsaklis and Sadrzadeh (2014). These two datasets show an intermediate level of grammatical complexity, as they involve complete sentences (while other datasets include smaller phrases), but have fixed length structures featuring similar syntactic constructions (i.e., transitive sentences). The two datasets differ with respect to size and construction method.

RELPRON consists of 1,087 pairs, split in development and test set, made up by a *target* noun labeled with a syntactic role (either *subject* or *direct object*) and a *property* expressed as *[head noun] that [verb] [argument]*. For instance, here are some example properties for the target noun *treaty*:

(1) a. OBJ treaty/N: document/N that delegation/N negotiate/V
b. SBJ treaty/N: document/N that grant/V independence/N

Transitive sentence similarity dataset consists of 108 pairs of transitive sentences, each annotated with human similarity judgments collected through the Amazon Mechanical Turk platform. Each transitive sentence in composed by a triplet *subject verb object*. Here are two pairs with high (2) and low (3) similarity scores respectively:

(2) a. government use power
b. authority exercise influence
(3) a. team win match
b. design reduce amount

3.2 Graph implementation

We tailored the construction of the DEG to this kind of simple syntactic structures, restricting it to the case of relations among pairs of event participants. Relations were automatically extracted from a 2018 dump of Wikipedia, BNC, and ukWaC corpora, parsed with the Stanford CoreNLP Pipeline (Manning et al., 2014).

Each $\langle (word_1, word_2), (r_1, r_2) \rangle$ pair was then weighted with a smoothed version of Local Mutual Information[1]:

$$LMI_\alpha(w_1, w_2, r_1, r_2) = f(w_1, w_2, r_1, r_2) log(\frac{\hat{P}(w_1, w_2, r_1, r_2)}{\hat{P}(w_1)\hat{P}_\alpha(w_2)\hat{P}(r_1, r_2)}) \quad (1)$$

where:

$$\hat{P}_\alpha(x) = \frac{f(x)^\alpha}{\sum_x f(x)^\alpha} \quad (2)$$

Each lexical node in DEG was then represented with its embedding. We used the same training parameters as in Rimell et al. (2016),[2], since we wanted our model to be directly comparable with their results on the dataset. While Rimell et al. (2016) built the vectors from a 2015 download of Wikpedia, we needed to cover all the lexemes contained in the graph and therefore we used the same corpora from which the DEG was extracted.

We represented each property in RELPRON as a triplet $((hn, r), (w_1, r_1), (w_2, r_2))$ where hn is the head noun, w_1 and w_2 are the lexemes that compose the proper relative clause, and each element of the triplet is associated with its syntactic role in the property sentence.[3] Likewise, each sentence of the transitive sentences dataset is a triplet $((w_1, nsbj), (w_2, root), (w_3, dobj))$.

3.3 Active Context implementation

In MEDEA, the SR is composed of two vectors:

- $\overrightarrow{LM}$, as the sum of the word embeddings (as this was the best performing model in literature, on the chosen datasets);
- $\overrightarrow{AC}$, obtained by summing up all the weighted centroids of triggered participants. Each *lexeme - syntactic role* pair is used to retrieve its 50 top s-neighbors from the graph. The top 20 re-ranked elements were used to build each weighted centroid. These threshold were choosen empirically, after a few trials with different (i.e., higher) thresholds (as in Chersoni et al. (2017b)).

We provide an example of the re-weighting process with the property *document that store maintains*, whose target is *inventory*: i.) at first the head noun *document* is encountered: its vector is activated as event knowledge for the *object* role of the sentence and constitutes the contextual information in AC against which GEK is re-weighted; ii.) *store* as a subject triggers some *direct object* participants, such as *product*, *range*, *item*, *technology*, etc. If the centroid were built from the top of this list, the cosine similarity with the target would be around 0.62; iii.) *s-neighbours* of *store* are re-weighted according to the fact that AC contains some information about the target already, (i.e., the fact that it is a document). The re-weighting process has the effect of placing on top of the list elements that are more similar to *document*. Thus, now we find *collection*, *copy*, *book*, *item*, *name*, *trading*, *location*, etc., improving the cosine similarity with the target, that goes up to 0.68; iv.) the same happens for *maintain*: its *s-neighbors* are retrieved and weighted against the complete AC, improving their cosine similarity with *inventory*, from 0.55 to 0.61.

3.4 Evaluation

We evaluated our model on RELPRON development set using Mean Average Precision (MAP), as

[1]The smoothed version (with $\alpha = 0.75$) was chosen in order to alleviate PMI's bias towards rare words (Levy et al., 2015), which arises especially when extending the graph to more complex structures than pairs.

[2]lemmatized 100-dim vectors with *skip-gram* with *negative sampling* (SGNS (Mikolov et al., 2013)), setting minimum item frequency at 100 and context window size at 10.

[3]The relation for the head noun is assumed to be the same as the target relation (either *subject* of *direct object* of the relative clause).

in Rimell et al. (2016). We produced the compositional representation of each property in terms of SR, and then ranked for each target all the 518 properties of the dataset portion, according to their similarity to the target. Our main goal was to evaluate the contribution of event knowledge, therefore the similarity between the target vector and the property SR was measured as the sum of the cosine similarity of the target vector with the $\overrightarrow{LM}$ of the property, and the cosine similarity of the target vector with the $\overrightarrow{AC}$ cued by each property. As shown in Table 2, the full MEDEA model (last column) achieves top performance, above the simple additive model LM.

	RELPRON		
	LM	AC	LM+AC
verb	0,18	0,18	**0,20**
arg	0,34	0,34	**0,36**
hn+verb	0,27	0,28	**0,29**
hn+arg	0,47	0,45	**0,49**
verb+arg	**0,42**	0,28	0,39
hn+verb+arg	0,51	0,47	**0,55**

Table 2: The table shows results in terms of MAP for the development subset of RELPRON. Except for the case of verb+arg, the models involving event knowledge in AC always improve the baselines (i.e., LM models).

For the transitive sentences dataset, we evaluated the correlation of our scores with human ratings with Spearman's ρ. The similarity between a pair of sentences s_1, s_2 is defined as the cosine between their LM vectors plus the cosine between their EK vectors. MEDEA is in the last column of Table 3 and again outperforms simple addition.

	transitive sentences dataset		
	LM	AC	LM+AC
sbj	0.432	0.475	0.**482**
root	0.525	0.547	0.**555**
obj	0.628	0.537	0.**637**
sbj+root	0.**656**	0.622	0.648
sbj+obj	0.653	0.605	0.**656**
root+obj	0.732	0.696	0.**750**
sbj+root+obj	0.732	0.686	0.**750**

Table 3: The table shows results in terms of Spearman's ρ on the transitive sentences dataset. Except for the case of sbj+root, the models involving event knowledge in AC always improve the baselines. *p-values* are not shown because they are all equally significant ($p < 0.01$).

4 Conclusion

We provided a basic implementation of a meaning composition model, which aims at being incremental and cognitively plausible. While still relying on vector addition, our results suggest that distributional vectors do not encode sufficient information about event knowledge, and that, in line with psycholinguistic results, activated GEK plays an important role in building semantic representations during online sentence processing.

Our ongoing work focuses on refining the way in which this event knowledge takes part in the processing phase and testing its performance on more complex datasets: while both RELPRON and the transitive sentences dataset provided a straight forward mapping between syntactic label and semantic roles, more naturalistic datasets show a much wider range of syntactic phenomena that would allow us to test how expectations jointly work on syntactic structure and semantic roles.

References

Nicholas Asher, Tim Van de Cruys, Antoine Bride, and Márta Abrusán. 2016. Integrating Type Theory and Distributional Semantics: A Case Study on Adjective–Noun Compositions. *Computational Linguistics*, 42(4):703–725.

Marco Baroni, Raffaela Bernardi, and Roberto Zamparelli. 2014. Frege in Space: A Program of Compositional Distributional Semantics. *Linguistic Issues in Language Technology*, 9(6):5–110.

William Blacoe and Mirella Lapata. 2012. A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 546–556. Association for Computational Linguistics.

Emmanuele Chersoni, Alessandro Lenci, and Philippe Blache. 2017a. Logical metonymy in a distributional model of sentence comprehension. In *Sixth Joint Conference on Lexical and Computational Semantics (* SEM 2017)*, pages 168–177.

Emmanuele Chersoni, Enrico Santus, Philippe Blache, and Alessandro Lenci. 2017b. Is structure necessary for modeling argument expectations in distributional semantics? In *12th International Conference on Computational Semantics (IWCS 2017)*.

Bob Coecke, Stephen Clark, and Mehrnoosh Sadrzadeh. 2010. Mathematical foundations for a compositional distributional model of meaning. Technical report.

Jeffrey L Elman. 2011. Lexical knowledge without a lexicon? *The mental lexicon*, 6(1):1–33.

Katrin Erk and Sebastian Padó. 2008. A structured vector space model for word meaning in context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 897–906. Association for Computational Linguistics.

Peter Hagoort and Jos van Berkum. 2007. Beyond the sentence given. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 362(1481):801–811.

Mary Hare, Michael Jones, Caroline Thomson, Sarah Kelly, and Ken McRae. 2009. Activating event knowledge. *Cognition*, 111(2):151–167.

Dimitri Kartsaklis and Mehrnoosh Sadrzadeh. 2014. A study of entanglement in a categorical framework of natural language. In *Proceedings of the 11th Workshop on Quantum Physics and Logic (QPL)*. Kyoto, Japan.

Gina R Kuperberg and T Florian Jaeger. 2016. What do we mean by prediction in language comprehension? *Language, cognition and neuroscience*, 31(1):32–59.

Alessandro Lenci. 2018. Distributional Models of Word Meaning. *Annual Review of Linguistics*, 4:151–171.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

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.

Ken McRae and Kazunaga Matsuki. 2009. People use their knowledge of common events to understand language, and do so as quickly as possible. *Language and linguistics compass*, 3(6):1417–1429.

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.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition.

Laura Rimell, Jean Maillard, Tamara Polajnar, and Stephen Clark. 2016. Relpron: A relative clause evaluation data set for compositional distributional semantics. *Computational Linguistics*, 42(4):661–701.

Richard Socher, Christopher D Manning, and Andrew Y Ng. 2010. Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*, volume 2010, pages 1–9.

LatInfLexi: an Inflected Lexicon of Latin Verbs

Matteo Pellegrini
Università di Bergamo/Pavia
Piazza Rosate, 2 –
24129 Bergamo, Italy
matteo.pellegrini@unibg.it

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

Abstract

English. We present a paradigm-based inflected lexicon of Latin verbs built to provide empirical evidence supporting an entropy-based estimation of the degree of uncertainty in inflectional paradigms. The lexicon contains information on the inflected forms that occupy the 254 morphologically possible paradigm cells of 3,348 verbal lexemes extracted from a frequency lexicon of Latin. The resource also includes annotation of vowel length and the frequency of each form in different epochs.

Italiano. *Presentiamo un lessico di forme flesse basato sui paradigmi per i verbi latini, costruito per fornire evidenza empirica che permetta di quantificare il grado di incertezza nei paradigmi flessivi tramite l'entropia. Il lessico contiene informazioni sulle forme flesse che occupano le 254 celle possibili dal punto di vista morfologico di 3.348 lessemi verbali estratti da un dizionario frequenziale del latino. La risorsa include anche l'annotazione della lunghezza vocalica e la frequenza di ogni forma in diverse epoche*.

1 Introduction

In this paper, we describe the construction of LatInfLexi, an inflected lexicon of Latin verbs organized in lexemes[1] and paradigm cells.

In morphological theory, there is a recent trend towards a more realistic modelling of complex inflectional systems: for instance, Ackerman et al. (2009) and Bonami and Boyé (2014) propose that the analysis should take a full inflected form as a starting point, without assuming any segmentation *a priori*. In such approaches, what is investigated is not the construction of forms from smaller units like stems and inflectional endings, but rather their predictability given knowledge of other forms. This can be done by using the information theoretic notion of conditional entropy to estimate the uncertainty in guessing the content of the paradigm cell of a lexeme knowing another inflected form of the same lexeme, by weighting the probability of application of each inflectional pattern based on their type frequency in real data.

To do so, large-scale inflected lexicons listing all forms of a representative selection of lexemes are needed. Such resources are increasingly being developed for modern languages – see among else Zanchetta and Baroni (2005) and Calderone et al. (2017) for Italian, Neme (2013) for Arabic, Bonami et al. (2014) and Hathout et al. (2014) for French. However, to the best of our knowledge, there are no resources of this kind for Latin, although their (semi-)automatic building is made possible by the current availability of several morphological analyzers for Latin, including *Words* (http://archives.nd.edu/words.html), Lemlat (www.lemlat3.eu), *Morpheus* (https://github.com/tmallon/morpheus), the PROIEL Latin morphology system (https://github.com/mlj/proiel-

[1] The term "lexeme" is used for the abstract theoretical concept normally adopted in morphology and lexicology, while "lemma" refers to the concrete citation form representing an entry in dictionaries. Since we aim at a resource suitable for theoretical inquiries, we use the first term as a label in our resource.

webapp/tree/master/lib/morphology) and *LatMor* (http://cistern.cis.lmu.de). Our resource was created to fill this gap and to enable a quantitative, entropy-based analysis of Latin verb inflection.

2 Design

A distinctive feature of our inflected lexicon is that it is based on lexemes and paradigm cells, rather than on forms. This means that for each lexeme, all the morphologically possible paradigm cells are filled with a form, and not only those forms that are indeed attested in Latin texts are stored in paradigm cells. In this respect, our resource is similar to other recently developed inflected lexicons, like for instance Flexique for French (Bonami et al., 2014).

For each paradigm cell, the following information is provided:

(i) the inflected form that occupies the paradigm cell;
(ii) a univocal identifier of the lexeme to which it belongs;
(iii) the set of its morphological features;
(iv) information on the frequency of the form in different epochs.

As for (i), it should be noted that there is never more than one form per paradigm cell. In cases of overabundance (i.e. cells that are filled by more than one form, cf. Thornton, 2012), a choice was made to decide which "cell-mate" (Thornton, 2012: 183) should be kept, and which one discarded.

On the other hand, in some cases a paradigm cell could be empty, either because it is defective – like for instance the passive cells of intransitive verbs – or because it is not filled by a synthetic form, but rather it is analytically expressed, by means of a phrase – like for instance, in Latin, the perfective cells of deponent verbs, for which the periphrasis PRF.PTCP[2] + AUX *esse* 'to be' is used (e.g. PRF.IND.1SG *hortātus sum* 'I incited'). In both cases, the cell is marked as #DEF# in the resource. This convention is adopted also in Flexique (Bonami et al., 2014: 2585), and it fits the requirements of the Qumin package for entropy calculations on the predictability of implicative relations between inflected forms (Bonami and Beniamine, 2016; Beniamine, 2017).

As for (ii), the identifier corresponds to the citation form of the lexeme, almost always the first-person singular of the present indicative, following the Latin lexicographical and didactical tradition. A diacritic is added in those rare cases where different verbs have the same citation form (see *infra*, §3.2).

Regarding (iii), we use the PoS-tags of the Universal Part-of-Speech Tagset by Petrov et al. (2012) and the morphological features used in Universal Dependencies (http://universaldependencies.org/u/feat/index.html).

Lastly, the frequency data in (iv) are taken from Tombeur's (1998) *Thesaurus Formarum Totius Latinitatis* (see *infra*, §3.3).

3 Building the Lexicon

This section details the procedure followed to build the lexicon.

3.1 Selecting the Lexemes

Our first objective is to build an inflected lexicon of Latin featuring all the possible inflected forms of verbs only. To this aim, we include all the verbal entries contained in Delatte et al.'s (1981) *Dictionnaire fréquentiel et Index inverse de la langue latine* (henceforth DFILL). This yields a total of 3,348 verbs. In rare cases, more than one entry of DFILL corresponds to one and the same lexeme in our resource. This happens because some verbs are lemmatized twice in DFILL. For instance, for the verb *verso* two different entries appear in DFILL, using as citation form both the first-person singular of the present active indicative *verso* and the corresponding morphologically passive form *versor*. This choice is likely to be motivated by the different semantics of the two verbs, with the first one meaning 'to turn' and the second one meaning 'to remain'. However, in such cases our resource gives priority to collecting into one common inflectional paradigm all the forms that can be assigned to the same lexeme based on their morphological relatedness, rather than separating them in paradigms of different lexemes according to semantic criteria. Therefore, our lexicon includes only one lexeme *verso*, for which both active and passive forms are listed.

[2] Throughout the paper, we will refer to grammatical features by using the standard abbreviations of the Leipzig Glossing Rules.

3.2 Generating the Forms

In order to fill all of the paradigm cells of the selected lexemes, we exploit the database of Lemlat (Passarotti et al., 2017). For each lexeme, the database of Lemlat contains a list of segments called LES – roughly corresponding to the stems that are used in different subparadigms – each with a corresponding CODLES that provides (among else) information on the inflectional endings that can be attached to a LES. We make use of this information to generate the relevant forms.

To illustrate the details of the procedure, let's consider the verb *rumpo* 'to break'. For this verb, the database of Lemlat features the LESs and CODLESs shown in Table 1.

LES	CODLES
rump	v3r
rumpisse	fe
rup	v7s
rupsit	fe
rupt	n41
rupt	n6p1
ruptur	n6p2

Table 1: the verb *rumpo* in Lemlat 3.0

The two LESs with CODLES "fe" ("forma eccezionale", 'exceptional form') were discarded, since they are full irregular forms that are stored as such. As for the other LESs, the one with CODLES "v3r" is used to fill all the cells of the present system, by adding the inflectional endings of the conjugation represented by the CODLES (i.e. the 3rd conjugation). Similarly, the LES with CODLES "v7s" is used to fill the cells of the perfect system. From the remaining LESs, some nominal forms built upon the so-called "third stem" (Aronoff, 1994) can be derived, namely the supine *rupt-um* and *rupt-ū* from the LES with CODLES "n41", the perfect participle *rupt-us, -a, -um* from the LES with CODLES "n6p1" and the future participle *ruptūr-us*, *-a, -um* from the LES with CODLES "n6p2".

This given, our first step is to extract information on the LESs and CODLESs of each lexeme. Since Lemlat is a tool built to analyze rather than produce forms, it contains also several LESs occurring only in irregular and/or rare forms. To avoid the risk of overgeneration, we choose and keep only one LES for each CODLES. The choice is based on lexicographical sources, namely Lewis and Short (1879) and Glare (1982). In these dictionaries, at the very beginning of each verbal entry there is a set of four "principal parts" (Bennett, 1908: 55), i.e. exemplary inflected forms from which the whole paradigm of the lexeme can be inferred. We keep only those LESs that correspond to such principal parts, excluding the ones that correspond to more marginal forms that do appear in dictionaries but are given less prominence in the entry. For instance, Lemlat includes two LESs with CODLES "v3r" for the verb *dico* 'to say': "dic" and "deic". However, in both the lexicographical sources we use, the relevant principal parts are *dico* and *dicere*, corresponding to the first LES, while the second one is only mentioned later in the entries as an alternative form. Therefore, the LES selected for our resource is "dic".

We use the same dictionaries also to manually annotate the vowel length for each LES. This is a necessary enhancement, because in Latin verb inflection there are homographic forms that can be distinguished only based on that, like for instance PRS.ACT.IND.3SG *fugit* '(s)he flees' vs. PRF.ACT.IND.3SG *fūgit* '(s)he fleed'.

Following this process, we fill all the 254 paradigm cells of each of the 3,348 lexemes. However, because of Lemlat's design, for some quite frequent verbs with a highly irregular inflectional paradigm, it was not possible to apply the same procedure, at least for the cells of the present system, which is where most irregularity of the inflectional endings of Latin verbs happens. For the verbs shown in Table 2 and for those derived from them by prefixation (e.g. *abeo* 'to go away' from verb *eo* 'to go'), although it was technically possible to adopt a similar approach by using more than one LES for a CODLES, it proved to be faster and practical to manually record the correct forms as such.

Lemma	Meaning
aio	to say
eo	to go
fero	to bring
fio	to become
inquam	to say
malo	to prefer
nolo	not to want
possum	can
sum	to be
volo	to want

Table 2: irregular verbs

To each of the 850,392 generated paradigms cells, a univocal lexeme identifier is assigned,

which corresponds to the lemma used in Lemlat. In those rare cases where two or more verbs have the same lemma in Lemlat (although they inflect differently), a numeric diacritic is added to make the relevant distinction: for instance, we have *volo1* 'to fly' and *volo2* 'to want'.

3.3 Frequency Data

Many forms included in the paradigm cells of our lexicon are never attested in Latin texts. In order to make it possible to distinguish between plausible but unattested forms and those indeed occurring in texts, we enhance forms with information on their frequency. This information is taken from Tombeur's (1998) *Thesaurus Formarum Totius Latinitatis* (henceforth TFTL), where each form is assigned the number of its occurrences in four different epochs, respectively called *Antiquitas* (from the origins to the end of the 2nd century A.D.), *Aetas Patrum* (2nd century-735 A.D.), *Medium Aeuum* (736-1499) and *Recentior Latinitas* (1500-1965).

By including the frequency of each form in the lexicon, we know how many of the 752,537[3] forms recorded in the lexicon are never actually attested. Table 3 reports the relevant data[4].

TFTL epoch	unattested forms (%)
Antiquitas	544,395 (72.34%)
Aetas Patrum	482,324 (64.1%)
Medium Aeuum	484,421 (64.37%)
Recentior Latinitas	640,552 (85.12%)
all epochs	401,690 (53.38%)

Table 3: not attested forms

It can be observed that a significant amount of forms recorded in our lexicon are not attested, even in such a large corpus as the one the TFTL is based on. However, this is not surprising: recent large-scale corpus-based investigations (e.g. Bonami and Beniamine, 2016: 158 ff.) show that in languages with large inflectional paradigms – like the ones of Latin verbs – it is perfectly normal that many plausible forms do not appear, even in very large datasets, and the lexemes for which the full paradigm is attested are very few.

4 Discussion and Future Work

We described the design and building of a lexeme-based inflected lexicon consisting of 850,392 paradigm cells of 3,348 Latin verbs. Our first objective in the near future is to make the resource complete in terms of lexical coverage, including the lexemes of the other PoS. The lexicon is available for download as a .csv file at `https://github.com/matteo-pellegrini/LatInfLexi`.

We also plan to include phonetic annotation, by giving the IPA transcription of each form, which can be obtained semi-automatically by applying a script provided by the Classical Language Toolkit (Johnson et al., 2014-17) to stems and endings.

Another welcome addition would be to account for cases of overabundance, by allowing more than one form to appear in the same paradigm cell. However, to decide which cell-mates to keep and which ones to discard, their frequency in Latin texts should be preliminarily evaluated. In this respect, it has to be noted that the frequencies in the TFTL refer to bare surface forms, with no contextual disambiguation. For instance, the frequency of *veniam* comprises not only occurrences of both the PRS.ACT.SBJV.1SG and FUT.ACT.IND.1SG of the verb *venio* 'to come', but also of the ACC.SG of the noun *venia* 'indulgence'.

To get an idea of the impact of morphological ambiguity on our lexicon, we analyzed all the generated forms with Lemlat (version 3.0). We found that only for about 23% (170,735) of the 752,537 forms Lemlat outputs only one analysis (i.e. one lemma and one set of morphological features), the remaining 581,802 (about 77%) being ambiguous. This result weakens the reliability of the frequency data provided in the lexicon. Therefore, disambiguation is needed, although this would require a very time-consuming work.

However, to tackle the problem of ambiguity, a first useful step is distinguishing between cases like *veniam* above, which can be analyzed as an inflected form of two different lemmas, and cases where the different analyses only refer to different forms of the same lemma, e.g. *laudatis*, that appears both in the PRS.ACT.IND.2PL and in

[3] The 97,855 paradigm cells marked as #DEF# are excluded from this count.

[4] In total, the TFTL includes 554,828 different forms, corresponding to 62,922,781 occurrences in the reference corpus used by the Thesaurus. Our lexicon contains 165,898 of these unique forms (forms appearing in more than one paradigm cell are counted only once), for a total of 18,261,179 occurrences. This means that our resource covers around 30% of the forms of the TFTL, in terms of both type and token frequency. In addition, it also contains several other forms that are not attested in the TFTL (245,623 unique forms).

the PRF.PTCP.DAT/ABL.PL of *laudo* 'to praise', but cannot be a form of other lemmas. We call these different types 'exolemmatic' and 'endolemmatic' ambiguity, respectively (cf. Passarotti and Ruffolo, 2004). Cases of exolemmatic ambiguity are clearly more problematic, but they are also much rarer: only 79,490 (about 10%) of the forms in our resource belong to this type. The great majority of ambiguous forms only give rise to endolemmatic ambiguity, as can be observed in Table 4 below, where the relevant data are summarized.

	n.	%
unambiguous forms	170,735	22.69%
ambiguous forms	581,802	77.31%
only endolemmatic amb.	502,312	66.75%
exolemmatic amb.	79,490	10.56%

Table 4: the impact of ambiguity on frequency data

As far as endolemmatic ambiguity is concerned, although its quantitative impact is far greater, it could be considerably reduced in a principled manner. Indeed, it should be noted that in many cases this kind of ambiguity is due to systematic syncretism. For instance, the cells FUT.ACT.IMP.2SG and FUT.ACT.IMP.3SG are never unambiguously analyzed, because they are always identical for a same verb. Given the full systematicity of this syncretism, which holds for all lexemes, these cells could be considered as only one from a purely morphological point of view. Therefore, the problem of endolemmatic ambiguity could be at least reduced by adopting an approach based on "morphomic paradigms" (Boyé and Schalchli, 2016), where always syncretic cells are conflated, rather than on morphosyntactic paradigms. This would be helpful especially in nominal forms like participles and gerundives, where such cases of systematic syncretism are widespread.

When such ambiguity issues will have been resolved, it will also be possible to exploit the frequency data in a more systematic fashion, e.g. to perform diachronic investigations on how the frequency of specific (groups of) forms or paradigm cells change across the four considered epochs, or to model Latin inflectional morphology in an even more realistic way, by considering also the token frequency of inflected forms, as has been recently proposed by Boyé (2016).

References

Farrell Ackerman, James P. Blevins, and Robert Malouf. 2009. Parts and wholes: Implicative patterns in inflectional paradigms. In James P. Blevins and Juliette Blevins, editors, *Analogy in Grammar: Form and Acquisition*. Oxford University Press, Oxford: 54–82.

Mark Aronoff. 1994. *Morphology by itself: Stems and inflectional classes*. MIT Press, Cambridge/London.

Sacha Beniamine. 2017. Un algorithme universel pour l'abstraction automatique d'alternances morpho-phonologiques. In *24e Conférence sur le Traitement Automatique des Langues Naturelles (TALN)*.

Charles Edwin Bennett. 1908. *New Latin Grammar*. Bolchazy-Carducci Publishers.

Olivier Bonami and Sarah Beniamine. 2016. Joint predictiveness in inflectional paradigms. *Word Structure* 9(2): 156–182.

Olivier Bonami and Gilles Boyé. 2014. De formes en thèmes. In Florence Villoing, Sophie David and Sarah Leroy, editors, *Foisonnements morphologiques: Études en hommage à Françoise Kerleroux*. Presses universitaires de Paris Ouest, Paris: 17–45.

Olivier Bonami, Gauthier Caron and Clément Plancq. 2014. Construction d'un lexique flexionnel phonétisé libre du français. In Franck Neveu, Peter Blumenthal, Linda Hriba, Annette Gerstenberg, Judith Meinschaefer and Sophie Prévost, editors, *Actes du quatrième congrès mondial de linguistique française*: 2583–2596.

Gilles Boyé. 2016. Pour une modélisation surfaciste de la flexion. Le cas de la conjugaison du français. In *SHS Web of Conferences*. Vol. 27. EDP Sciences.

Gilles Boyé and Gauvain Shalchli. 2016. The status of paradigms. In Andrew Hippisley and Gregory Stump, editors, *The Cambridge Handbook of Morphology*. Cambridge University Press, Cambridge: 206–234.

Basilio Calderone, Matteo Pascoli, Nabil Hathout and Franck Sajous. 2017. Hybrid method for stress prediction applied to GLAFF-IT, a large-scale Italian lexicon. In *International Conference on Language, Data and Knowledge*. Springer, Cham: 26–41.

Louis Delatte, Étienne Evrard, Suzanne Govaerts and Joseph Denooz. 1981. *Dictionnaire fréquentiel et index inverse de la langue latine*. L.A.S.L.A, Liege.

Peter G.W. Glare. 1982. *Oxford Latin Dictionary*. Oxford University Press, Oxford.

Nabil Hathout, Franck Sajous and Basilio Calderone. 2014. GLÀFF, a large versatile French lexicon. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*: 1007–1012.

Kyle P. Johnson et al. 2014-2017. *CLTK: The Classical Language Toolkit*. DOI 10.5281/zenodo593336.

Charlton Lewis and Charles Short. 1879. *A Latin Dictionary*. Clarendon, Oxford.

Alexis Amid Neme. 2013. A fully inflected Arabic verb resource constructed from a lexicon of lemmas by using finite-state transducers. *Revue RIST: revue de l'information scientifique et technique* 20(2): 7–19.

Marco Passarotti, Marco Budassi, Eleonora Litta and Paolo Ruffolo 2017. The Lemlat 3.0 Package for Morphological Analysis of Latin. In *Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language*: 24–31.

Marco Passarotti and Paolo Ruffolo. 2004. L'utilizzo del lemmatizzatore LEMLAT per una sistematizzazione dell'omografia in latino. *EUPHROSYNE* 32(A): 99–110.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2011. A universal part-of-speech tagset. *ArXiv*:1104–2086

Anna M. Thornton. 2012. Reduction and maintenance of overabundance. A case study on Italian verb paradigms. *Word Structure* 5(2): 183–207.

Paul Tombeur. 1998. *Thesaurus formarum totius latinitatis a Plauto usque ad saeculum XXum*. Brepols, Turnhout.

Eros Zanchetta and Marco Baroni. 2005. Morph-it!: a free corpus-based morphological resource for the italian language.

Word Embeddings in Sentiment Analysis

Ruggero Petrolito•, Felice Dell'Orletta◇
• Università di Pisa
ruggero.petrolito@gmail.com
◇Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC–CNR)
ItaliaNLP Lab - *www.italianlp.it*
felice.dellorletta@ilc.cnr.it

Abstract

English. In the late years sentiment analysis and its applications have reached growing popularity. Concerning this field of research, in the very late years machine learning and word representation learning derived from distributional semantics field (i.e. word embeddings) have proven to be very successful in performing sentiment analysis tasks. In this paper we describe a set of experiments, with the aim of evaluating the impact of word embedding-based features in sentiment analysis tasks.

Italiano. *Recentemente la Sentiment Analysis e le sue applicazioni hanno acquisito sempre maggiore popolarità. In tale ambito di ricerca, negli ultimi anni il machine learning e i metodi di rappresentazione delle parole che derivano dalla semantica distribuzionale (nello specifico i word embedding) si sono dimostrati molto efficaci nello svolgimento dei vari compiti collegati con la sentiment analysis. In questo articolo descriviamo una serie di esperimenti condotti con l'obiettivo di valutare l'impatto dell'uso di feature basate sui word embedding nei vari compiti della sentiment analysis.*

1 Introduction

In the late years sentiment analysis has reached great popularity among NLP tasks. As reported by Mäntylä et al. (2016) the number of papers on this subject has increased significantly in the first two decades of 21st century, as well as the extent of its applications. A wide variety of technologies has been used to assess sentiment analysis tasks during this period. In the latter years, machine learning techniques proved to be very effective; in particular, in recent years systems based on deep learning techniques represent the state of the art. In this field, **word embeddings** have been widely used as a way of representing words in sentiment analysis tasks, and proved to be very effective.

A relevant mirror of the state of the art in sentiment analysis field can be found in the *SemEval* workshops. In the 2015 edition (Rosenthal et al., 2015), most participants used machine learning techniques; in many of the subtasks, the top ranking systems used deep learning methods and word embeddings, like the system submitted by Severyn and Moschitti (2015), which was ranked 1st in subtask A and 2nd in subtask B. In 2016 edition (Nakov et al., 2016), deep learning based techniques, such as convolutional neural networks and recurrent neural networks, were the most popular approach. In 2017 edition (Rosenthal et al., 2017), machine learning methods were very popular, especially support vector machines and deep neural networks like convolutional neural networks and long short-term neural networks.

Concerning Italian language, **EVALITA** conference well represents the state of the art in the natural language processing field. In 2016 edition (Barbieri et al., 2016), the top ranking systems used machine learning and deep learning techniques (Castellucci et al. (2016), Attardi et al. (2016), Di Rosa and Durante (2016)).

The purpose of this study is to explore ways of using word embeddings to build meaningful representations of documents in sentiment analysis tasks performed on Italian tweets.

2 Our Contribution

In this paper we aimed to evaluate the effect of exploiting word embeddings in sentiment analysis tasks. In particular, we explore the effect of five factors on the performance of a sentiment analysis classification system, to answer five research questions:

1. What is the effect of the size of the corpus used to train the embeddings?
2. Which text domain allows us to train better embeddings (in-domain vs out-of-domain data)?
3. Which type of learning method produces better embeddings (word vs character-based word embeddings)?
4. Which method to combine the word vectors produces a better document vector representation?
5. What are the most important words (in terms of part-of-speech) to produce a better document vector representation?

To answer such questions, we performed several classification experiments testing our system on the three sentiment analysis tasks proposed in the 2016 EVALITA SENTIPOLC campaign (Barbieri et al., 2016): **Subjectivity Classification**, **Polarity Classification** and **Irony Detection**. In the first of these tasks, the highest accuracy was achieved by the system of Castellucci et al. (2016). Concerning the 2nd task, the most accurate system was the one submitted by Attardi et al. (2016). Regarding the 3rd task, the highest accuracy value was reached by the system of Di Rosa and Durante (2016). Among these systems, Castellucci et al. (2016) and Attardi et al. (2016) use deep learning techniques (convolutional neural networks), while Di Rosa and Durante (2016) use an ensemble of many supervised learning classifiers.

3 Datasets

We tested our system on the three sentiment analysis tasks proposed in 2016 EVALITA SENTIPOLC campaign. These tasks and the related datasets have been described by Barbieri et al. (2016). We conducted our experiments on the training set provided by the organizers of the evaluation campaign, which is composed of 7921 tweets.

We train our word embeddings on two corpora: in-domain and out-domain. The in-domain dataset is a collection of tweets that we collected for this work, named **Tweets**. It is composed by almost 80 millions of tweets, resulting in around 1.2 billions of tokens. The out-of-domain dataset is the **Paisà** corpus, a collection of Italian web texts described by Lyding et al. (Lyding et al., 2013).

4 Experimental Setup

For our experiments, we used a classifier based on SVM using LIBLINEAR (Rong-En et al., 2013) as machine learning library. As features, the classifier uses only information extracted combining the word-embeddings of the words of the analyzed tweet.

In all the experiments described in this paper, our system addresses the classification tasks by performing **5-fold cross-validation** on the training set provided for the SENTIPOLC 2016 evaluation campaign. The final score is the average score. We evaluate each fold using the Average F-score described by Barbieri et al. (2016).

For what concerns the word embeddings, we trained two types of word embedding representations: i) the first one using the **word2vec**[1] toolkit (Mikolov et al., 2013). This tool learns lower-dimensional word embeddings, which are represented by a set of latent (hidden) variables, and each word is associated to a multidimensional vector that represents a specific instantiation of these variables; ii) the second one using **fastText** (Bojanowski et al., 2016), a library for efficient learning of word representations and sentence classification. This library allows to overcome the problem of out-of-vocabulary words which affects the methodology of word2vec. Generating out-of-vocabulary word embeddings is a typical issue for morphologically rich languages with large vocabularies and many rare words. FastText overcomes this limitation by representing each word as a bag of character n-grams. A vector representation is associated to each character n-gram and the word is represented as the sum of these character n-gram representations.

In both cases, each word is represented by a 100 dimensions vector, computed using the CBOW algorithm – that learns to predict the word in the middle of a symmetric window based on the sum of the vector representations of the words in the window – and considering a context window of 5 words.

5 Experiments and Results

To answer the questions listed in Section 2, we conducted a great amount of experiments, testing many ways of representing the tweets by exploiting in different manners the word embeddings of

[1] http://code.google.com/p/word2vec/

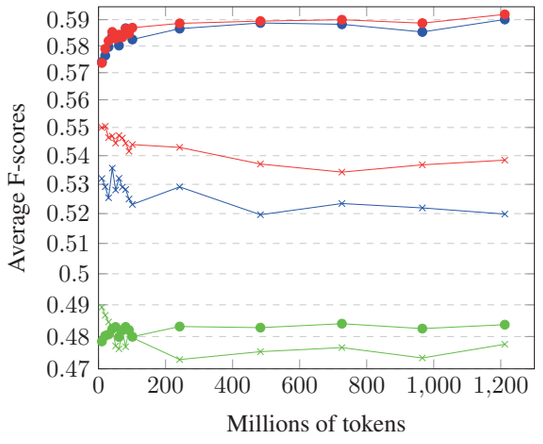


Figure 1: Average F-scores obtained by using embeddings trained on increasing amounts of token, using word2vec (circles) and fastText (crosses). Blue is assigned to *Subj. Classification*, red to *Pol. Classification* and green to *Irony Detection*.

the words extracted from the tweets.

To evaluate the impact (in terms of classification accuracy) of the variations of each studied parameter, we report the accuracy for each variation of the parameter calculated as the average accuracy across all the classification experiments that we conducted by varying all the other parameters (in a 5-fold cross-validation scenario).

In all the experiments, we used only features based on word embeddings.

5.1 Size of the Embeddings Training Corpus

To answer the question n. 1, we trained several word embedding models on different partitions of *Tweets* corpus of increasing sizes, using both *word2vec* and *fastText*. Ten smaller partitions were obtained starting with just ten millions of tokens (for the smaller one) and adding other ten millions for each new partition, reaching the amount of 100 millions. We created other four bigger partitions, which contain respectively 240, 480, 720 and 960 millions of tokens; the size of the smaller of this four partitions is comparable to the size of *Paisà*.

Figure 1 reports the results. When we use embeddings trained with word2vec on increasing amounts of data, the average value of F-score grows for all the three subtasks. The amount of this growth is similar for the subtasks *Subjectivity Classification* (0.016) and *Polarity Classification* (0.019), while it's smaller for the subtask *Irony Detection*, which is the most challenging among the three. In all cases the increase is significantly faster in the first 80 to 100 millions of tokens, particularly as regards the *Irony Detection* task: in this case, the average F-score basically stops growing after around 80 millions of tokens.

When we use embeddings trained with fastText, the outcome is the opposite: the average F-score values decrease as bigger amounts of data are used to train the embeddings. The decrease of the values is faster when using the first hundreds of millions of tokens.

Lesson learned: these results suggest that, regarding word-based word embeddings, as the training corpus grows the accuracy rises, but it becomes stable quickly. On the other hand, the increase of the size of the training corpus apparently doesn't influence the accuracy values when the embedding have been produced using fastText (or it even causes a lowering of the accuracy values).

5.2 Domain of the Embeddings Training Corpus

To answer the question n. 2, we ran a set of experiments using the four models obtained using word2vec and fastText on **Paisà** and **Tweet** corpora. Table 1 reports the results of the experiments. As we can see, the embeddings trained with word2vec on the in-domain dataset (*Tweets*) provide features that allow to achieve a higher average accuracy compared to the features extracted from the out-domain corpus. Differently, there isn't any variation in terms of accuracy when the embeddings are trained with fastText.

Lesson learned: the in-domain word embeddings are very important in a semantic classification scenario. Apparently, this is not true when character-based word embedding are used.

	Subj.		Pol.		Iro.	
	w2v	ft	w2v	ft	w2v	ft
tw	**0.5901**	0.5198	**0.592**	0.5384	**0.4837**	0.4776
pa	0.572	0.5206	0.5693	0.5312	0.4793	0.4759

Table 1: Average F-scores obtained by using word embeddings trained on Twitter (tw) and Paisà (pa) corpora.

5.3 Type of Embeddings Learning Model

For what regards the question n. 3, the type of embeddings learning model (words vs character n-grams) influences considerably the performance of the classifier. Using embeddings trained with word2vec leads to F-score values that are significantly higher in comparison to the accuracy ob-

tained using embeddings trained with fastText (see Table 1).

Lesson learned: this outcome suggests that embeddings learned by methods that treat words as atomic entities provide features that are more useful in a semantic task such as sentiment classification, in comparison with character-based embeddings.

5.4 Methods to Combine Word Embeddings

To answer the question n. 4, we tested many methods to combine the embeddings of the words of each document into a document-level vector representation.

We experimented five combining methods: *Sum*, *Mean*, *Maximum-pooling*, *Minimum-pooling*, *Product*. Each of this methods returns a single vector $\vec{t}$, such that each t_n is obtained by combining the nth components $w_{1n}, w_{2n} \dots w_{mn}$ of the embedding of each tweet word. Figure 2 shows a graphical representation of this process.

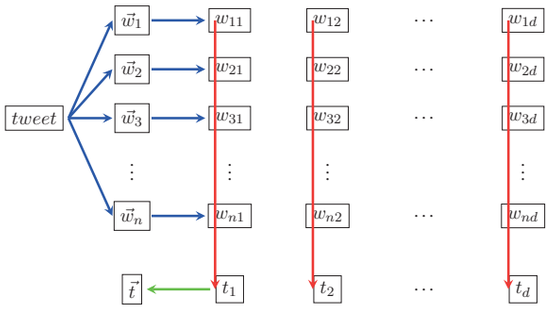


Figure 2: Embeddings combination process

We tested these methods separately, and all of them jointly as well. When using all methods, the document representation is obtained concatenating the vectors returned by each method.

As we can see in Table 2, the *Sum* method proved to be the best method for all the tasks, when using embeddings obtained by word2vec. The best results overall are obtained using the concatenation of each of the vectors returned by the used methods (row *All* in the Table). When using embeddings trained with fastText, the best results are obtained with *mean* for *Subjectivity* and *Polarity Classification*, and with *sum* for *Irony Detection*. In this case, the combination of all vector leads to poor results.

Lesson learned: these outcomes suggest that the best combination methods are *sum* for word vectors obtained by using word-based word embeddings and *mean* for character-based ones.

	Subj.		Pol.		Iro.	
	w2v	ft	w2v	ft	w2v	ft
Sum	**0.6054**	0.534	**0.6085**	0.5532	**0.4887**	**0.5033**
Mean	0.6017	**0.5951**	0.5954	**0.5916**	0.4709	0.4811
Max	0.5957	0.5012	0.5964	0.507	0.4736	0.4698
Min	0.593	0.5012	0.5951	0.5011	0.4754	0.4707
Prod	0.4415	0.4759	0.4384	0.5012	0.4693	0.4628
All	**0.6236**	0.4846	**0.6246**	0.51	**0.5202**	0.4715

Table 2: Average F-scores obtained by using different strategies of combination of word embeddings. Bold black values are the best F-scores overall; blue bold values are the best F-scores obtained by using a single combination method in the word-based word embeddings scenario (w2v); red bold values are the best F-scores in the character-based word embeddings scenario (ft).

Meanwhile, the worst approach is the *Product* combination. Interestingly, while the concatenation of all the combined word-based word embeddings is surely the best approach to produce the document-level vector representation, this is not true for the character-based ones.

5.5 Selection of Morpho-syntactic Categories of Combined Word Embeddings

To answer the question n. 5, we ran a set of experiments using only a subset of the word embeddings of each document to produce the document vector representation. The word selection is guided by the morpho-syntactic categories of the words. We tested four categories: *noun*, *verb*, *adjective*, *adverb*. The embeddings of the words belonging to each of these categories were combined in a pos-based vector representation document. In addition, we tested the document representation vector obtained through the concatenation of the different pos-based vectors (*N, V, Adj, Adv*) with and without the all-word document vector *All words*, which is the only one taking into account emoticons and hash tags.

Table 3 reports the results of the experiments. In the word-based word embedding scenario, regarding the contribution of single morpho-syntactic categories, *noun* shows the highest performance. Overall, the highest score is yielded by the combination of all the selected categories concatenated with the combined vector of all the word embeddings (*All words* rows in the table). For what regards the character-based word embeddings, we can see that the *noun* is the individually best performing category only for the Subjectivity Classification task, while the *adjective* and the *verb* are the best performing category for the other two tasks.

	Subj.		Pol.		Iro.	
	w2v	ft	w2v	ft	w2v	ft
N	**0.553**	**0.5171**	**0.5417**	0.5091	**0.4725**	0.4749
V	0.4755	0.4778	0.5091	0.5136	0.469	**0.4897**
Adj	0.4406	0.4534	0.5184	**0.5335**	0.4705	0.4826
Adv	0.4397	0.4504	0.4971	0.5033	0.4702	0.485
N, V, Adj, Adv	0.6266	0.5578	0.6141	0.5667	0.4948	0.5041
All words	0.6251	0.5363	0.5941	0.515	0.4773	0.4521
All words, N	0.6287	0.5221	0.6032	0.5343	0.4887	0.4646
All words, V	0.6326	0.5276	0.6035	0.5339	0.4841	0.4634
All words, Adj	0.6374	0.5328	0.6185	0.5184	0.4867	0.4693
All words, Adv	0.6337	0.5243	0.6087	0.5187	0.4856	0.4674
All words, N, V, Adj, Adv	**0.6521**	0.5691	**0.6319**	0.5546	**0.5139**	0.4886

Table 3: Average F-scores obtained using embedding of words belonging to different morpho-syntactic classes. Bold black values are the best F-scores overall; blue bold values are the best F-scores obtained using a single grammar class in the word-based word embeddings scenario (w2v); red bold values are the best F-scores obtained using a single grammar class in the character-based word embeddings scenario (ft).

Lesson learned: these results show that *noun* class is the most important grammatical category only in the word-based word embedding scenario; meanwhile the concatenation of all the pos-based vectors and the *All words* vector yields the best accuracy in both scenarios.

6 Conclusions

In this work we study the impact of word embedding-based features in the sentiment analysis tasks. We performed several classification experiments to investigate the effects on classification performances of five dimensions related to the word embeddings. We tested several different ways of selecting and combining the embeddings and we studied how the performance of a sentiment classifier changes.

Despite the lessons learned from this work, several aspects remain to investigate, such as, for example, the tuning of the parameters used to train the embeddings, and new vector combining strategies.

References

Giuseppe Attardi, Daniele Sartiano, Chiara Alzetta and Federica Semplici. 2016. *Convolutional Neural Networks for Sentiment Analysis on Italian Tweets*. CLiC-it/EVALITA.

Francesco Barbieri, Valerio Basile, Danilo Croce, Malvina Nissim, Nicole Novielli and Viviana Patti. 2016. *Overview of the evalita 2016 sentiment polarity classification task*. 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).

Yoshua Bengio, Réjean Ducharme, Pascal Vincent and Christian Jauvin. 2003. *A Neural Probabilistic Language Model*. Journal of Machine Learning Research 3 (2003) 1137–1155.

Piotr Bojanowski, Edouard Grave, Armand Joulin and Tomas Mikolov. 2016. *Efficient Estimation of Word Representations in Vector Space*. CoRR abs/1607.04606, 2016.

Giuseppe Castellucci, Danilo Croce and Roberto Basili. 2016. *Context-aware Convolutional Neural Networks for Twitter Sentiment Analysis in Italian*. CLiC-it/EVALITA.

Emanuele Di Rosa and Alberto Durante. 2016. *Tweet2Check evaluation at Evalita Sentipolc 2016*. CLiC-it/EVALITA.

Verena Lyding, Egon Stemle, Claudia Borghetti, Marco Brunello, Sara Castagnoli, Felice Dell'Orletta Henrik Dittmann, Alessandro Lenci and Vito Pirrelli. 2013. *PAISÀ Corpus of Italian Web Text*. Institute for Applied Linguistics, Eurac Research.

Mika V. Mäntylä, Daniel Graziotin and Miikka Kuutila. 2016. *The Evolution of Sentiment Analysis - A Review of Research Topics, Venues, and Top Cited Papers*. Computer Science Review, Volume 27, February 2016, Pages 16-32.

Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean. 2013. *Efficient Estimation of Word Representations in Vector Space*. CoRR abs/1301.3781, 2013.

Preslav Nakov, Alan Ritter, Sara Rosenthal, Fabrizio Sebastiani and Veselin Stoyanov. 2016. *SemEval-2016 task 4: Sentiment analysis in Twitter*. Proceedings of the 10th international workshop on semantic evaluation (semeval-2016).

Fan Rong-En, Chang Kai-Wei, Hsieh Cho-Jui, Wang Xiang-Ruind Lin Chih-Jen. 2008. *LIBLINEAR: A Library for Large Linear Classification*. Journal of Machine Learning Research, 9:1871-1874.

Sara Rosenthal, Preslav Nakov, Svetlana Kiritchenko, Saif Mohammad, Alan Ritter and Veselin Stoyanov. 2015. *Semeval-2015 task 10: Sentiment analysis in twitter*. Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015), 451–463.

Sara Rosenthal, Noura Farra and Preslav Nakov. 2017. *SemEval-2017 task 4: Sentiment analysis in Twitter*. Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), 502–518.

Aliaksei Severyn and Alessandro Moschitti. 2015. *Unitn: Training deep convolutional neural network for twitter sentiment classification*. Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015), 464–469.

Identifying Citation Contexts: a Review of Strategies and Goals.

Agata Rotondi, Angelo Di Iorio, Freddy Limpens
Department of Computer
Science and Engineering
University of Bologna, Italy
agata.rotondi@unibo.it
angelo.diiorio@unibo.it
freddy.limpens@unibo.it

Abstract

English. The Citation Contexts of a cited entity can be seen as little tesserae that, fit together, can be exploited to follow the opinion of the scientific community towards that entity as well as to summarize its most important contents. This mosaic is an excellent resource of information also for identifying topic specific synonyms, indexing terms and citers' motivations, i.e. the reasons why authors cite other works. Is a paper cited for comparison, as a source of data or just for additional info? What is the polarity of a citation? Different reasons for citing reveal also different weights of the citations and different impacts of the cited authors that go beyond the mere citation count metrics. Identifying the appropriate Citation Context is the first step toward a multitude of possible analysis and researches. So far, Citation Context have been defined in several ways in literature, related to different purposes, domains and applications. In this paper we present different dimensions of Citation Context investigated by researchers through the years in order to provide an introductory review of the topic to anyone approaching this subject.

Italiano. *Possiamo pensare ai Contesti Citazionali come tante tessere che, unite, possono essere sfruttate per seguire l'opinione della comunità scientifica riguardo ad un determinato lavoro o per riassumerne i contenuti più importanti. Questo mosaico di informazioni può essere utilizzato per identificare sinonimi specifici e Index Terms nonchè per individuare i motivi degli autori dietro le citazioni. Identificare il Contesto Citazionale ottimale è il primo passo per numerose analisi e ricerche. Il Contesto Citazionale è stato definito in diversi modi in letteratura, in relazione a differenti scopi, domini e applicazioni. In questo paper presentiamo le principali dimensioni testuali di Contesto Citazionale investigate dai ricercatori nel corso degli anni.*

1 Introduction and Background

Researchers consider as Citation Context (CC) different snippets of text around a citation marker. These differences of width influence the applications that exploit CC as source of information. For example, Qazvinian and Radev (2010) showed that using also implicit citations (i.e. sentences that contain information about a specific secondary source but do not explicitly cite it) for generating surveys, rather than citing sentences alone, improve the results. Ritchie et al. (2008) compared different widths of CC in order to find the most appropriate window for identifying Index Terms. They proved that varying the context from which the Index Terms are gathered has a significant effect on retrieval effectiveness. Aljaber et al. (2010) tested different sizes of CC for a document clustering experiment. They claimed that a window size of 50 words from either side of the citation marker works better than taking 10 or 30 terms or the citing sentence alone, whatever its size is. From their analysis, relevant synonymous and related vocabulary extracted from this window of text, in combination with an original full-text representation of the cited document, are effective for document clustering. We can claim that the issue of finding the optimal CC for a specific application is a challenging task that interests researchers and which is at the base of every study that exploits the CC as a source of information.

Usage | Type of CC	Fixed Number of Characters	Citing Sentence	Extended Context
Content Reading	CiteSeerX, Knoth et al. (2017)	Semantic Scholar	Research Gate[fixed],, Fujiwara and Yamamoto (2015)[fixed],
Automatic Summary (article summary, domain surveys, background information extraction, etc.)		Elkiss et el. (2008)	Mei and Zhai (2008)[fixed], Nanba and Okumura (1999)[adaptive], Qazvinian and Radev (2010)[adaptive]
Article Ranking / Clustering / Indexing / Searching / Bibliometrics	Bradshaw (2003), Aljaber et al. (2010), Doslu and Bingol (2016)		J. O'Connor (1992)[adaptive],
Semantic Interpretation of Articles		Nakov et al. (2004)	
Sentiment Analysis / Citation Functions		Sula and Miller (2014), Bertin et al. (2016)	Athar and Teufel 2012[adaptive], Abu-Jbara et al 2013[adaptive]
Citation Context Analysis			Kaplan et al. (2009)[adaptive], Kaplan et al. (2016)[adaptive], Abu-Jbara and Radev (2012)[citation scope]
Pros	Easy to implement, doesn't need domain and linguistic analysis, available extraction tool (Parscit)	Easy to implement (it just needs a good sentence tokenizer), provides a processable content (e.g. by linguistic parsers)	**FixedEC**: easy to implement, includes more information **AdaptiveEC**: clean and complete result (especially if combined with citation scope identification)
Cons	Risk to include noise or to miss citation information	Risk to include noise or to miss citation information	**FixedEC:** Risk to include noise **AdaptiveEC**: Challenging domain specific implementation

Figure 1: Survey Summary

1 With the purpose of providing a useful background to anyone approaching this question, in the following sections we give an overview of different dimensions of textual CC investigated in literature. We classified them in 3 main categories: a) fixed number of characters b) citing sentence c) extended context (fixed and adaptive), and we summarized our analysis in Figure1. We focus on the strategies to identify the correct textual CC of a citation, nevertheless other CC related topics have been investigated in literature as for example citation recommendations (see Farber (2018) and Ebesu (2017))

The belief of the need of a clear introductory survey about how CC has been differently shaped in literature came to our mind when we faced the problem of defining the optimal CC for the Semantic Coloring of Academic References (SCAR) project[1] (Di Iorio et al., 2018). The goal of the SCAR project is to enrich bibliographies of scientific articles by adding explicit meta data about individual bibliographic entries and to characterize these entries according to multiple criteria. With this purpose, we are studying a set of properties to support the automatic characterization of bibliographic entries and one of our primary source of information is the textual content around citation markers, i.e. the CC. We are currently investigating on finding the best span of text for our needs. By reviewing the literature, we realized that different approaches correspond to different tasks and are also related to the linguistic domain of application. The SCAR project as well as this review are focused on the English language but it would be interesting to extend this study to other languages.

[1] http://dasplab.cs.unibo.it/index.php/scar/

2 Fixed Number of Characters

A good way to start exploring how the CC can be diversely defined is to look for well known examples. One of these is the public search engine and digital library for scientific and academic papers CiteSeerX[2]. This web platform allows users to browse papers' references and to read the context in which a reference is cited. The function enables the reading of 200 characters before and after the citation marker. Here the choice of the CC width is not directly related to further analysis and applications as the purpose is the mere reading of text by users. As Ii et al. (2014) describe, CiteSeerX uses ParsCit (Councill et al., 2008) for citation extraction. ParsCit is a freely available, open-source implementation of a reference string parsing package which performs reference string segmentation and CC extraction. The size of the context is configurable, but by default extends to 200 characters on either side of the match. ParsCit is a well know software and is used in different projects. For example, the Association Of Computational Linguistics (ACL) Anthology Network[3] uses ParsCit for curation. Doslu and Bingol (2016) also used ParsCit in their work regarding how to rank articles for a given topic. The authors exploited the information contained in the CC of a certain paper for detecting important articles and providing focused directions to access the literature about a topic. They stated that the words that are used to describe a cited paper stand close to the citation marker, and this is their motivation for choosing a fixed window size context. Before Doslu and Bingol, also Bradshaw (2003) used CC to index cited

[2] http://citeseerx.ist.psu.edu/index

[3] http://aan.how/index.php/home/about

paper for specific topics. He designed the Reference Direct Indexing in which measures of relevance and impact are joined in a single retrieval metric based on the comparison of the terms authors use in multiple CC of a document. The CC Bradshaw used to index the documents are directly gathered from CiteSeerX. Also the tool presented by Knoth et al. (2017), who address the problem of automatically retrieving and collecting CC for a given unstructured research paper, extract a CC window of fixed length corresponding to 300 characters before and after a citation marker. The approach of considering as CC a fixed length snippet around the citation marker is a naive baseline method. It can be used to retrieve terms related to a cited entity and the accuracy of applications that employ it might be improved for example by considering sentence or paragraph boundaries(Aljaber et al., 2010). This kind of context is unsuitable if the CC needs to be further analyzed, for example by using syntactic parsers, or if its content have to be represented in a coherent formal way where the meaning and structure of sentences have to be preserved.

3 Citing Sentence

Another famous platform among scholars is Semantic Scholar[4]. This subjective search service for journal articles provides several functions for browsing papers among which the possibility of quickly read the CC of each citation. This service allows reading more than one excerpt of text for each entity (when available). Each CC shown corresponds exactly to a citing sentence, i.e. the sentence that contains the targeted reference marker. Implicit citations[5] are also investigated by exploiting lexical hooks and also in these cases the CC excerpts shown are in the form of a full sentence. The same CC window has been adopted in several projects. Nakov et al. (2004) investigated the use of CC for semantic interpretation of bioscience articles. Starting from the collection of the citing sentences related to a specific cited entity (that they call *citances*), they used the output of a dependency parser to build paraphrase expressing relations between two named entities. As commented before, parsers need to be fed with full sentences in order to provide proper representations and this work is a clear example where a fixed length CC would not have been an appropriate input. Also Elkiss et al. (2008) focused their research on the set of citing sentences of a given article (named by the authors *citation summaries*) testing the biomedical domain. Despite Elkiss study did not rely on any strictly sentence based technique (they employed cosine similarity and tf-idf), both their hypothesis are grounded on the importance of citing sentences boundaries. Sula and Miller (2014) presented an experimental tool for extracting and classifying citation contexts in humanities. Their approach is based on citing sentences from which they extracted features (e.g. location in document) and polarity (evaluating n-grams with a naive Bayes classifier). Bertin et al. (2016) followed a similar approach to identify n-grams and sentiment in CC. They chose to work on a sentence basis stating that sentences are the natural building blocks of text and likely to include the context of a specific reference. Starting from citing sentences they extracted 3-grams containing verbs, together with position in the paper and type of section according to the IMRaD structure in order to analyze the combination and distribution of these features in the biomedical domain. Citing sentence as a base unit for CC is mostly chosen in hard sciences domains. In fact, scientific communities have particular ways of using language and specific conventions that reveal clear disciplinary differences. Hyland (2009) describes some of these language variations that go from terminology differences to different citations practices and rhetorical preferences. Writers use different sets of reporting verbs to refer to others work (engineers *show*, philosophers *argue*, biologists *find* and linguists *suggest*); frequencies of hedges and self citations, directives and n-grams also diverge across fields. In the humanities writers tend to include extensive referencing and build a background for the heterogeneous readership while in hard sciences most of the readers share a common context with writers. This attitude clarifies citers' behaviors in different domains and makes us presume that CC in humanities might be more complex than in hard sciences. Following these considerations, it is reasonable to con-

[4]https://www.semanticscholar.org

[5]More in details, with implicit citations we refer to those mentions of a work where the relation cited entity-citing entity is not provided by a citation marker but rather by a lexical object related to the cited entity. E.g.: *The heuristics based on **WordNet** and Wikipedia ontologies are very sensitive to preprocessing* is an implicit citation of George A. Miller (1995). WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11: 39-41.

clude that for choosing the appropriate CC width one needs to take into account not only the task he is going to face but also the domain of applications and the specificity of the language. In this sense, CC as citing sentence might not always correspond to the entire fragment of text referring to a targeted citation marker.

4 Extended Context

Extending CC beyond the citing sentence can prove useful in many cases as illustrated by the social networking site for researchers ResearchGate[6]. Every document in this platform's database can be inspected according to different prospectives. Among them, readers can browse documents citations lists and access CC (when available) displayed in the form of: 1 sentence before the citing sentence + citing sentence + 1 sentence after the citing sentence. This window size allows users to better understand the full context of a citation without loosing any possible informations contained in the nearby sentences. This is particularly relevant for the task of polarity identification of citations. Athar and Teufel (2012) have shown that authors' sentiments are most likely expressed outside the citing sentences. Sentiment in citations is often hidden and especially criticism might be hedged both for politeness and for political reasons (MacRoberts and MacRoberts, 1984). Citing sentences are typically neutral and in particular negative polarity occurs in the following sentences (Teufel et al., 2006), see for example (from (Platt, 1990)):

In [19, sec. 11.11], Vapnik suggests a method for mapping the output of SVM to probabilities by decomposing the feature space []. Preliminary results for this method, are ***promising****.However, there are some* ***limitations*** *that are overcome by the method of this chapter.*

Particularly for, but not limited to, polarity identification tasks, a context extended to the nearby sentences can supply the complete set of information about a citation to applications and readers. Sentences nearby a citing sentence can be add as part of the CC according to a fixed schema or by following an adaptive approach.

[6]https://www.researchgate.net

4.1 Fixed Extended Context

Besides ResearchGate and the aforementioned Ritchie's work, who studied different window sizes of CC for identifying Index Terms, also Mei and Zhai (2008) implemented a fixed extended context for their study of summarizing articles influence. For their impact-based summarization task they used a 5 sentences window size, with 2 sentences before and after the citing sentence. This technique allows to include more info in the CC but at the same time the risk of adding noise is high. This is why most of the literature concerning extended CC rather provides adaptive methods.
A mention is needed to the work of Fujiwara and Yamamoto (2015), mostly for their overall project than for the CC retrieval approach which relies on a very basic technique (they include the sentence after the citing one if the reference marker is at the end of the citing sentence and limit long citing sentences to 240 characters before and after citation markers). The authors built the Colil database where CC of the life sciences domain are stored, and made it available to users through a web-based search service. For each resource stored in the database, a list of CC in which the resource has been cited is returned to the user who can easily read how a work is perceived and used by different authors.

4.2 Adaptive Extended Context

O'Connor (1982) was the first who investigated the CC as a sequence of sentences - a multi-sentence citing statement. His purpose was to study the words of CC as possible improvement for the retrieval of the related cited entities. He wrote 16 complex and detailed computer rules (not completely computer procedures at that time) with linguistic, structural and more general features for the selection of citing statements. Nanba and Okumura (1999) presented a system to support writing surveys of a specific domain. They see the CC as a succession of sentences where the possible connections are indicated by 6 kinds of cue words (anaphora, negative expression, 1st and 3rd person pronoun, adverb, other) that they use for retrieving the suitable CC for their system. To identify the full span of CC, Kaplan et al. (2009) presented a different method based on co-reference chains. They built a SVM (Cortes and Vapnik, 1995) classifier with 13 features (among which: cosine similarity, gender and number agreement,

semantic class agreement etc.) that are tested in order to find the best configuration. Results of the classifier alone and in combination with cue-based techniques are promising. Despite the little data analyzed for the project, Kaplan raised some interesting remarks about CC. Particularly, they stated that sentences of CC are not necessarily contiguous. Qazvinian and Radev (2010) explored the task of retrieving background information close to explicit citations by implementing a probabilistic inference model (Markov Random Field). Like previous authors, they observed that the majority of sentences related to a citation directly occur after or before the citation or another context sentence; however they also confirmed Kaplan's intuition about possible gaps between sentences describing a cited paper. Athar and Teufel (2012) tried to go further by attempting to retrieve all the mentions of a cited entity within the full text of the citing paper. As claimed by the authors, mentions to a cited entity can occur in the full article and are necessary to identify the real sentiment toward the cited work. Their first experiment of manual annotation proved the insight that retrieving all the mentions of a cited entity increases citation sentiment coverage. Also the SVM framework implemented by the authors, despite limited to a 4 sentence window, outperformed a single sentence baseline system. Abu-Jbara et al. (2013), with the purpose of adding qualitative aspects to standard quantitative bibliometrics (H-Index, G-Index, etc.), analyzed the text surrounding a citation in order to define the citer's purposes and polarity. This piece of text (CC), is retrieved with a sequence labeling method. Starting from the citing sentence, Abu-Jbara's team used CRF (Lafferty et al., 2001) to determine if the sentence before and the two sentences after the citing sentences have to be included in the CC. The features for the CRF model are both structural (e.g. position of the current sentence with respect to the citing sentence) and lexical (e.g. presence of demonstrative determiners). Kaplan et al. (2016) named Citation Block Determination(CBD) the task of detecting non-explicit citing sentences and faced it by testing various features representing different aspects of textual coherence. Non local mentions are excluded from what they formalized as a binary classification task of sentences from the citing one. They tested different relational and entity coherence features and their combinations. Experiments showed that the CRF method fits better the task than the SVM approach.

The different works briefly described so far give an overview of the most interesting techniques explored by researchers. From rule-based approaches to probability methods, the implemented features are most of the time domain-specific relying on particular vocabulary and on stylistic and rhetorical habits.

4.2.1 Citation Scope

Related to the Adaptive Extended Context topic is the identification of the Scope of a citation. So far we have discussed different ways of including in the CC what is outside the citing sentence but at the same time related to it. The idea is to extend the context. However, there are cases in which the citing sentence does not completely refer to the targeted citation or where the context of multiple citations overlap. In these cases the aforementioned approaches of CC extraction would include noise and affect applications results. See for instance the following example where the whole citing sentence might produce a negative polarity despite the neutral value of the citation:

The negative results produced by the BoW approach led our team to change direction and we tested a SVM(CORTES, 1995) classifier.

Finding a procedure to cut out the precise scope of a citation is a tricky and challenging task for which little experiments have been done.

Athar (2011) suggested to trim the parse tree of each citing sentence and to keep only the deepest clause in the subtree of which the citation is a part. Abu-Jbara and Radev (2012) explored 3 different methods for identifying the scope: word classification, sequence labeling and segment classification. Results showed that the scope of a given reference consists of units of higher granularity than words. In fact, the segment classification technique achieved the best performance. Despite the interesting results, we agree with Hernandez and Gomez (2016) who stated that additional work is required to improve the citation scope identification task. The need of further research in this field is also encouraged by the analysis of Jha et al. (2017) who performed an annotation experiment on a sample of the ACL Anthology Network revealing that, on average, the reference scope for a given target reference contains only 57.63 per

cent of the original citing sentence.

5 Conclusion

We have reviewed what we consider the most interesting works about CC identification in order to provide a solid background to anyone interested in the topic and especially to those researchers who are facing the task of identifying the best approach for their studies. We did not compare the different strategies with the purpose of ranking them, but we rather showed that there exists various relations between a methodology and the usage, domain, and language specificity of its possible applications.

References

Abu-Jbara, A., and Radev, D. 2012. *Reference Scope Identification in Citing Sentences*. In Proc. of NAACL HLT, (p. 80-90).

Abu-Jbara, A., Ezra, J., and Radev, D. 2013. *Purpose and Polarity of Citation: Towards NLP-based Bibliometrics*. In Proc. of NAACL HLT, (p. 596-606).

Aljaber, B., Stokes, N., Bailey, J., and Pei, J. 2010. *Document Clustering of Scientific Texts Using Citation Contexts*. Information Retrieval, 13, (p.101-131).

Athar, A. 2011. *Sentiment Analysis of Citations using Sentence Structure-Based Features*. Proceedings of NAACL-HLT, (p.81-87).

Athar, A., and Teufel, S. 2012. *Detection of Implicit Citations for Sentiment Detection.*. In Proc. of DSSD, (p. 18-26).

Bertin, M., Atanassova, I., Sugimoto, C., and Lariviere, V. 2016. *The Linguistic Patterns and Rhetorical Structure of Citation Context: an Approach Using N-Grams*. Scientometrics, 109(3).

Bradshaw, S. 2003. *Reference Directed Indexing: Redeeming Relevance for Subject Search in Citation Indexes*. In Proc. of ECDL, (p. 499-510).

Cortes, C. and Vapnik V. 1995. *Support-Vector Networks*. Machine Learning, 20 (3), (p. 273-297).

Councill, I., Giles, C., and Kan, M. 2008. *ParsCit : An Open-Source CRF Reference String Parsing Package*. In Proc. of LREC, (p. 661-667).

Di Iorio, A., Limpens, F., Peroni, S., Rotondi, A., and Tsatsaronis, G. 2018. *Investigating Facets to Characterise Citations for Scholars*. In Proc. of SAVE-SD Workshop.

Doslu, M., and Bingol, H. 2016. *Context Sensitive Article Ranking with Citation Context Analysis*. Scientometrics, 108 (2), (p. 653671).

Ebesu, T., and Fang, Y. 2017. *Neural Citation Network for Context-Aware Citation Recommendation*. In Proc. of SIGIR, (p. 10931096).

Elkiss A., Shen S., Fader A., Erkan G., States D., and Radev D. 2008. *Blind Men and Elephants: What Do Citation Summaries Tell Us About a Research Article?*. American Society for Information Science and Technology, 59 (1), (p. 51-62).

Farber M., Thiemann A., and Jatowt A. 2018. *To Cite, or Not to Cite? Detecting Citation Contexts in Text*. In Proc. of ECIR: Advances in Information Retrieval, (p. 598-603).

Fujiwara, T., and Yamamoto, Y. 2015. *Colil: a Database and Search Service for Citation Contexts in the Life Sciences Domain*. Biomedical Semantics, 6(38).

Hernandez-Alvarez, M., and Gomez, J. 2016. *Survey About Citation Context Analysis: Tasks, Techniques, and Resources*. Natural Language Engineering, 22(3), (p. 327-349).

Hyland, K. 2009. *Writing in the Disciplines: Research Evidence for Specificity*. Taiwan International ESP Journal, 1(1), (p. 5-22).

Jha, R., Jbara, A., Qazvinian, V., and Radev D. 2017. *NLP-driven Citation Analysis for Scientometrics*. Natural Language Engineering, 23(1), (p. 93-130).

Lafferty, J., McCallum, A., and C.N. Pereira, F. 2001. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*. In Proc. of ICML, (p. 282-289).

Ii, A., Wu, J., and Giles, C. 2014. *CiteSeerX : Intelligent Information Extraction and Knowledge Creation from Web-Based Data*. In Proc. of AKBC, (p. 1-7).

Kaplan, D., Iida, R., and Tokunaga, T. 2009. *Automatic Extraction of Citation Contexts for Research Paper Summarization: A Coreference-Chain Based Approach*. In Proc. of NLPIR4DL, (p. 88-95).

Kaplan, D., Tokunaga, T., and Teufel, S. 2016. *Citation Block Determination Using Textual Coherence*. Information Processing, 24(3), (p. 540-553).

Knoth, P., Gooch, P. and Jack, K. 2017. *What Others Say About This Work? Scalable Extraction of Citation Contexts from Research Papers*. In Proc. of TPDL, (p. 287299).

Mei, Q., and Zhai, C. 2008. *Generating Impact-Based Summaries for Scientific Literature*. In Proc. of ACL-HLT, (p. 816-824).

MacRoberts, M.H., and MacRoberts, B.R. 1984. *The Negational Reference: or the Art of Dissembling*. Social Studies of Science, 14, (p. 91-94).

Nakov, P., Schwartz, A., and Hearst, M. 2004. *Citances: Citation Sentences for Semantic Analysis of Bioscience Text*. In Proc. of SIGIR.

Nanba, H., and Okumura, M. 1999. *Towards Multi-paper Summarization Using Reference Information*. In Proc. of IJCAI, (p. 926-931).

O'Connor, J. 1982. *Citing Statements: Computer Recognition and Use to Improve Retrieval*. Information Processing and Management, 18(3), (p. 125-131).

Platt J.C. 1990. *Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods*. Advances in Large Margin Classifiers, (p. 61-74).

Qazvinian, V., and Radev, D. 2010. *Identifying Non-explicit Citing Sentences for Citation-based Summarization*. In Proc. of ACL, (p. 555-564).

Ritchie, A., Robertson, S., and Teufel, S. 2008. *Comparing Citation Contexts for Information Retrieval*. In Proc. of ACM-CIKM, (p. 213-222).

Sula, C., and Miller, M. 2014. *Citations, Contexts, and Humanistic Discourse: Toward Automatic Extraction and Classification*. Literary and Linguistic Computing, 29, (p. 453-464).

Teufel, S., Siddharthan, A., and Tidhar, D. 2006. *Automatic Classification of Citation Function*. In Proc. of EMNLP, (p. 103-110).

DialettiBot: a Telegram Bot for Crowdsourcing Recordings of Italian Dialects

Federico Sangati
University L'Orientale
Naples, Italy
fsangati@unior.it

Ekaterina Abramova
Nijmegen University
The Netherlands
e.abramova@ftr.ru.nl

Johanna Monti
University L'Orientale
Naples, Italy
jmonti@unior.it

Abstract

English. In this paper we describe DialettiBot, a Telegram based chatbot for crowdsourcing geo-referenced voice recordings of Italian dialects. The system enables people to listen to previously recorded audio and encourages them to contribute to building a collective linguistic resource by sending voice recordings of their own spoken dialects. The project aims at collecting a large sample of voice recordings in order to promote knowledge of linguistic variation and preserve proverbs or idioms typical for different local dialects. Moreover, the collected data can contribute to several voice-based Natural Language Processing (NLP) applications in helping them understand utterances in non-standard Italian.

Italiano.

In questo articolo descriviamo DialettiBot, un chatbot basato su Telegram per raccogliere registrazioni audio geo-referenziate di dialetti italiani. Il sistema permette alle persone di ascoltare le registrazioni precedentemente inserite, e le incoraggia a contribuire alla costruzione di questa risorsa linguistica collettiva, attraverso l'invio di registrazioni audio nel proprio dialetto. Il progetto mira a raccogliere una grande mole di registrazioni che possono aiutare a promuovere la conoscenza delle variazioni linguistiche e la salvaguardia dei proverbi o modi di dire tipici di ogni dialetto locale. I dati raccolti possono inoltre contribuire a diverse applicazioni del trattamento automatico del linguaggio (TAL) che hanno bisogno di essere adattate per comprendere espressioni dialettali.

1 Introduction

It is commonly known that Italy has an abundance of different dialects, such as Florentine, Venetian, and Neapolitan. These dialects are not only characterized by simple phonetic variation as it is usually meant by this term, but they are proper Romance languages, with a fully developed grammar and lexicon. As Repetti puts it:

> The Italian 'dialects' [...] are daughter languages of Latin and sister languages of each other, of standard Italian, and of other Romance languages, and they may be as different from each other and from standard Italian as French is from Portuguese. (Repetti, 2000)

This dialectical variety is a resource that deserves to be studied and preserved for both cultural and applied reasons. The former, because it is quickly disappearing with less and less people who regularly use dialect at home and in public places. According to UNESCO "Atlas of the World's Languages in Danger",[1] there are about 2,500 endangered languages worldwide. In Italy, thirty dialects are at risk of extinction, such as friulano, ladino and veneciano.[2] The applied motivation is that in recent years we have witnessed a significant growth in the number of voice-based NLP applications (such as virtual assistants), which are currently not trained on local dialects and therefore perform poorly with a number of Italian speakers.

In this paper we present a freely available tool that enables geo-referenced recording of Italian dialects: *DialettiBot*, a Telegram based chatbot, whose aim is to collect a large sample of voice recordings, promoting preservation of linguistic

[1] http://www.unesco.org/languages-atlas

[2] http://www.culturaitalia.it/opencms/en/contenuti/focus/UNESCO_warns_that_thirty_Italian_dialects_are_at_risk_of_extinction.html?language=en

variation and its use in NLP applications. The rest of the paper is organized as follows: in section 2 we describe related work, in section 3 the implemented system and in section 4 the collected data.

2 Related work

There has been an extensive linguistic research of Italian dialects (Lepschy and Lepschy, 1992; Belletti, 1993; D'Alessandro et al., 2010). Here we summarize a number of projects that relate to the idea of gathering linguistic recordings for producing a map of dialects. We also point out their limitations that inspire our project.

VIVALDI project the "Vivaio Acustico delle Lingue e dei Dialetti d'Italia" is a collection of recordings and transcriptions of fixed phrases in the dialects of different cities from all regions in Italy (Kattenbusch et al., 1998). Unfortunately, the project is no longer active and has mainly focused on a finite set of chosen sentences, as opposed to spontaneous utterances.

LOCALINGUAL A web application for crowdsourcing recordings from around the world. This project is the one that most closely relates to ours. The main difference is that it is not restricted to a specific country, does not use geo-locations and works via a web application, which makes it difficult to be used on mobile devices or in case of poor data connection.[3]

ALF Atlas Linguistique de la France: an influential dialect atlas of Romance varieties in France published in 13 volumes between 1902 and 1910 (Gilliéron and Edmont, 1902). An example of more recent work of this type is Hall, Damien (2012).[4]

ALD Linguistic Atlas of Dolomitic Ladinian and neighbouring Dialects (Skubic, 2000). The project studies the linguistic variation between dialects of the region which covers the Grisons and Friuli region.[5]

IDEA The International Dialects of English Archive was created in 1998 as the internet's first archive of primary-source recordings of English-language dialects and accents as heard around the world. With roughly 1,400 samples from 120 countries and territories, and more than 170 hours of recordings, IDEA is now the largest archive of its kind.[6]

MICROCONTACT aims at developing a theory of syntactic change by observing the evolution of the dialects spoken by Italians who have migrated to North and South America during the 20th century.[7]

SPEAKUNIQUE and VOCALID are two similar projects that aim at collecting English voice sample from different regions for creating personalized digital voices for communication text to speech devices.[8]

Our project aims to be an updated and continuously evolving initiative that can capture spontaneous (living) dialectical variation over the whole Italian territory by being freely accessible and easy to use for a variety of non-specialists. As such, the project follows methodological practices similar to other citizen-science projects (Gurevych and Zesch, 2013; Simpson et al., 2014; Hosseini et al., 2014), it incorporates a GWAP[9] feature (Lafourcade et al., 2015), and fits within the line of 'explicit crowdsourcing' as defined by the EnetCollect[10] COST[11] action.

3 System description

In order to crowdsource recordings from Italian dialects, we have built a Telegram chatbot: DialettiBot.[12] As shown in the screenshot in figure 1, the user can interact with the bot via a standard dialogue chat interface in a Telegram application which is freely available for all mobile or desktop operating systems.[13] Apart from textual input, the interface provides a small keyboard of buttons that changes during the dialogue flow to simplify the interaction. In addition, the bot is able to accept vocal recordings and GPS locations.

The bot gives the possibility to the user to listen to approved recordings or to add new ones.

In the **listening mode**, it is possible to search for recording based on location or view the list

[3] https://localingual.com
[4] http://cartodialect.imag.fr/cartoDialect/accueil
[5] https://www.micura.it/en/activities/ald-linguistic-atlas
[6] https://www.dialectsarchive.com
[7] https://microcontact.sites.uu.nl/project
[8] https://www.speakunique.org, https://www.vocalid.co
[9] Game with a purpose.
[10] http://enetcollect.eurac.edu
[11] European Cooperation in Science and Technology.
[12] https://t.me/dialettibot
[13] https://telegram.org/apps

Figure 1: Screenshot of the DialettiBot system.

of the most recent recordings. As an element of gamification (Lafourcade et al., 2015), there is the possibility to ask for a random recording and try to guess its location. The user would then receive a feedback about the distance between the guessed location and the correct one. With this simple game we gather valuable data that would enable us to plot a type of confusability matrix between dialects, i.e., how much a dialect of place A resembles a dialect of place B.

In the **recording mode**, the user is asked to submit a freely chosen vocal recording of a sentence, that can be a simple phrase or a proverb, typical for their dialect. In addition, the user is asked to indicate the place where the dialect comes from (either by sending a GPS location or inputting the name of the place – in case the user is not currently located in the place associated with the dialect), and optionally the translation of the recording in Italian.

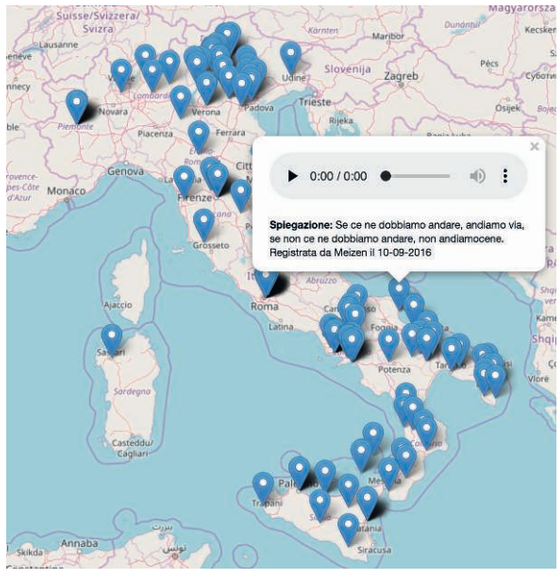


Figure 2: Screenshot of the web application displaying the audio map of the approved recordings.

As soon as the recording is submitted, the administrator of the system receives a notification (via the bot) with the new recording and is asked to approve or reject the contribution. Typical causes of rejection are too much background noise and explicitly offensive utterances. In case of approval, the recording is inserted in the database and becomes readily available to other users in the listening mode.[14]

In addition to the bot application, we developed a web application[15] (see figure 2) for visualizing the approved recordings in a map and giving the possibility to click on each of them to listen to the audio and read the translation.

3.1 Technical Specification

The bot is implemented in Python using the telegram bot API.[16] We chose to deploy the system via a chatbot (as opposed to a mobile app or web application) because it is much faster to build and to maintain since all the major functionalities (voice recordings, GPS location) are already embedded in the chat application and immediately accessible via simple API calls. Moreover, the system works on all mobile and desktop platforms without the need to build system-specific versions. Fi-

[14] In the future, there is a possibility to implement an additional validation step where other users or experts might flag some contribution as not being representative of a dialect.

[15] http://dialectbot.appspot.com/audiomap/mappa.html

[16] https://core.telegram.org/bots/api

nally, the simplified interface of a chatbot is particularly suitable to elderly people which are one of the most valuable target groups of the project, and can be easily used for recording other people while traveling also in case of no data connection (recordings are saved locally and uploaded to the server when data connection is again available).

The server behind DialettiBot is hosted by the Google Application Engine (GAE) framework and the data is stored in the integration datastore. The GAE technology guarantees full scalability up to an unrestricted number of users which could enable producing a significantly large volume of recordings. The same system also serves the web application with the map of the recordings illustrated in figure 2, which has been implemented in javascript using the *Leaflet*[17] library.

4 Collected data

The first version of DialettiBot has been deployed in January 2016. Since then, 1,886 users have interacted with the system and have submitted 255 voice recordings out of which 220 have been approved.[18] About 14% of users who interacted with the system contributed a recording.

Figure 3 shows the bar chart with the distribution of the approved recordings over time. The plot shows that the number of contributions in 2017 (31) has been significantly lower than in 2016 (117) , whereas in 2018 the number is increasing again (72 in the first 3 quarters of the year).

Figure 4 shows the distribution of the approved recordings on the map of Italy, with the counts clustered by proximity (heat map). Campania is the region with most recordings (38), followed by Lazio (35), Trentino-South Tyrol and Sicily (27), Puglia (22), Veneto (15), Piedmont and Tuscany (12), Calabria and Lombardy (9), Basilicata (5), Emilia-Romagna, Friuli-Venezia Giulia and Marches (2), Abruzzo, Molise and Sardinia (1). Currently we have no recordings from Liguria, Umbria and Valle d'Aosta.

5 Conclusions and future work

We have presented DialettiBot, a chatbot system based on Telegram for crowdsourcing georeferenced recordings of Italian dialects.

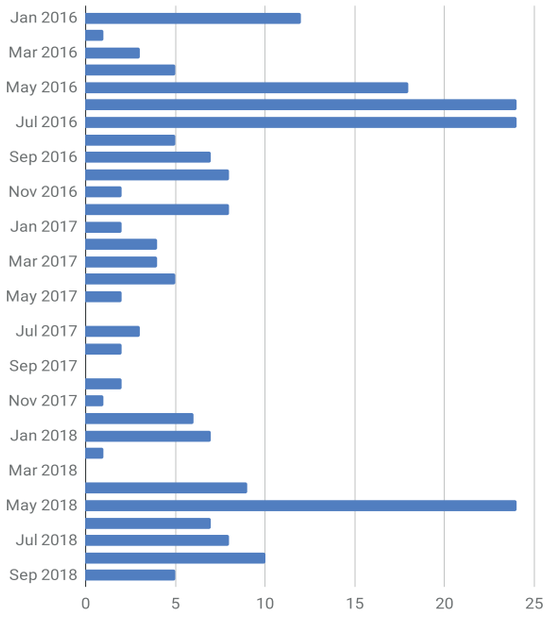


Figure 3: Frequency of approved recordings collected over time.

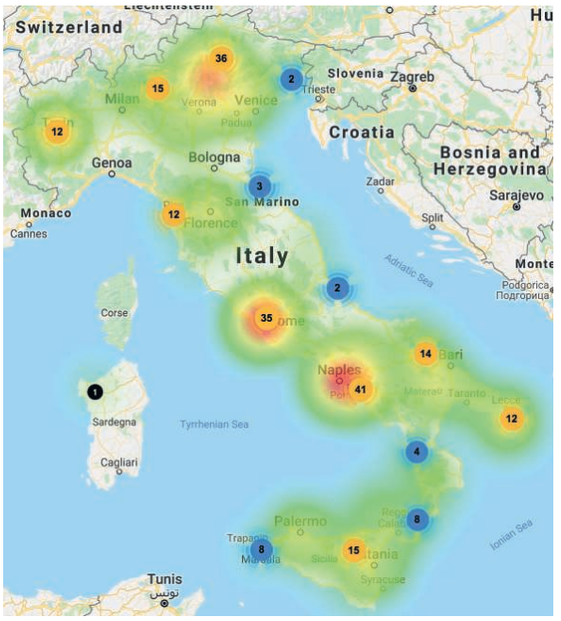


Figure 4: Heat map of the approved recordings.[19]

[17] https://leafletjs.com
[18] As of 31st of September 2018.
[19] Created via https://mapmakerapp.com.

Preliminary tests show that the system can be easily used by anyone who wishes to collect data in the field as well as the dialect speakers themselves. The recording quality is good and the data is easily exportable to be used for further processing in the service of linguistic research or NLP applications. At the same time, the current state of the project suffers from a number of limitations that need to be addressed in future work and that we discuss next.

First, the preliminary tests have not been informed by a detailed linguistic study of dialectical variation nor have we implemented a methodology for data collection. This is because the tests have been carried out as a proof-of-concept for the technology used to collect linguistic resources rather than a full-fledged linguistic project. Future tests will require a more careful consideration for dialect characteristics in the Italian language, the type of data that would be most valuable (spontaneous speech vs a set of set sentences etc.) and a construction of precise, reproducible instructions for the contributors.

Second, as described in section 3, we make use of a centralized validation procedure to approve a subset of recordings. However, since we have no complete knowledge of all Italian dialects we may end up accepting recordings which are not mapped to the correct location. In the future, we would like to decentralize the procedure, by delegating the approval to a higher number of volunteers spread out in all the regions, so that each new recording will get validated by the closest volunteer.

Finally, the number of users and recordings collected so far is relatively modest. This is due to the fact that no effort has been undertaken so far to promote its use by researchers or the general public. Accordingly, the current goal of the project is to get support from cultural institutions (both at a local and at a national level) to help us engage the citizens in this crowdsourcing effort, as well as academic partners to further refine the methodology and extend the chatbot capabilities.

We believe this project could contribute to help safeguard the Italian dialectic richness and collect useful resources for NLP applications, as we intend to make all recordings openly available for other researchers to use.[20]

[20]We are planning to upload the data to the Common Language Resources Infrastructure (CLARIN).

Acknowledgments

We kindly acknowledge all users who have so far contributed to the project by providing audio recordings of their dialects, and the three anonymous reviewers for their useful feedback.

References

A. Belletti. 1993. *Syntactic Theory and the Dialects of Italy*. Volume 9 of Linguistica (Turin, Italy). Rosenberg & Sellier.

Roberta D'Alessandro, Adam Ledgeway, Ian Roberts, and Frank Nuessel. 2010. *Syntactic Variation: The Dialects of Italy*. Cambridge University Press.

Jules Gilliéron and Ed. Edmont. 1902. *Atlas linguistique de la France,*. H. Champion,, Paris,.

Iryna Gurevych and Torsten Zesch. 2013. Collective intelligence and language resources: Introduction to the special issue on collaboratively constructed language resources. *Lang. Resour. Eval.*, 47(1):1–7.

Hall, Damien. 2012. Vers un nouvel atlas linguistique de la france. *SHS Web of Conferences*, 1:2171–2189.

Mahmood Hosseini, Keith Phalp, Jacqui Taylor, and Raian Ali. 2014. The four pillars of crowdsourcing: a reference model. IEEE Eighth International Conference on Research Challenges in Information Science.

Dieter Kattenbusch, Carola Köhler, Marcel Lucas Müller, and Fabio Tosques. 1998. VIVALDI project: Vivaio acustico delle lingue e di dialetti d'italia. https://www2.hu-berlin.de/vivaldi.

M. Lafourcade, A. Joubert, and N.L. Brun. 2015. *Games with a Purpose (GWAPS)*. Focus Series in Cognitive Science and Knowledge Management. Wiley.

A.L. Lepschy and G.C. Lepschy. 1992. *The Italian Language Today*. Hutchinson university library. Routledge.

Lori Repetti. 2000. *Phonological Theory and the Dialects of Italy*. John Benjamins Publishing Company.

Robert Simpson, Kevin R. Page, and David De Roure. 2014. Zooniverse: Observing the world's largest citizen science platform. In *Proceedings of the Companion Publication of the 23rd International Conference on World*

Wide Web Companion, WWW Companion '14, pages 1049–1054. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland.

Mitja Skubic. 2000. Ladinia linguistica in una monumentale opera: Atlante linguistico del ladino dolomitico e dei dialetti limitrofi - ald-1, dr. ludwig reichert verlag, wiesbaden 1998. *Linguistica*, 40(1):188–195.

Bootstrapping Enhanced Universal Dependencies for Italian

Maria Simi
Dipartimento di Informatica
Università di Pisa
Largo B. Pontecorvo 3, Pisa
simi@di.unipi.it

Simonetta Montemagni
Istituto di Linguistica Computazionale
"A. Zampolli" - CNR
Via Moruzzi 1, Pisa
simonetta.montemagni@ilc.cnr.it

Abstract

English. The paper presents an extension of the Italian Universal Dependencies Treebank with an "enhanced" representation level (e-IUDT), aimed at simplifying the information extraction process. The modules developed to semi-automatically build e-IUDT were delexicalized to perform cross-language enhancements: preliminary experiments in this direction led to promising results.

Italiano. *L'articolo presenta l'estensione della Universal Dependencies Treebank italiana (e-IUDT) con un livello di rappresentazione arricchito ("enhanced"), finalizzato a rendere più efficiente ed efficace il processo di estrazione dell'informazione. I moduli sviluppati per la costruzione semi-automatica della risorsa sono stati delessicalizzati e utilizzati per il trattamento di diverse lingue: esperimenti preliminari in questa direzione mostrano risultati promettenti.*

1 Introduction

The Universal Dependencies (UD) project, launched in 2015, aims at developing cross-linguistically consistent treebank annotation for many languages, with the goal of facilitating multilingual parser development, cross-lingual learning, and parsing research from a language typology perspective (Nivre et al., 2016). UD represents an open community effort with over 200 contributors producing more than 100 treebanks in over 60 languages.

Starting from the Stanford Dependencies project, from which Universal Dependencies (UD) originate, two syntactic representation options are made available, suited to different use cases (De Marneffe and Manning, 2008): the so-called "basic" representation where a close parallelism to the source text is maintained (i.e. where each word of the original sentence is present as a node), and the so-called "collapsed and propagated" representation which was conceived with a specific view to information extraction tasks.

Within the current version of UD, the "collapsed and propagated" representation has evolved into the graph-based *enhanced* representation proposed by Schuster and Manning (2016).

Since UD version 2.2 (officially released on July 2018), "enhanced treebanks" started to appear for a limited number of languages, i.e. English, Finnish, Russian, Polish, Dutch, Latvian. In order to foster the development of enhanced treebanks for other languages, transfer experiments exploiting existing treebanks are reported in the literature, following both rule-based (Schuster and Manning 2016) and data-driven (Nyblom *et al.*, 2013) approaches.

This paper describes the approach we used for developing and validating the enhanced version of the Italian UD Treebank and reports the first results of transfer experiments to English.

2 Enhanced dependencies

Enhanced dependencies were proposed as a way to simplify the process of information extraction. Enhancements, for the most part, result in additional links added to the dependency tree, motivated by inferences, which remain however anchored at the surface representation level. The result of enhancing a dependency tree is a graph, possibly with cycles, but not necessarily a super graph (since some of the original arcs may be discarded).

The current UD guidelines are quite conservative, i.e. they suggest practically feasible enhancements only. Despite this, enhancements cannot always be achieved automatically, and the task is challenging enough to be interesting. Ac-

cording to the guidelines *enhanced graphs* may contain some or all of the following enhancements, described with particular emphasis on Italian:

1. Added subject relations in control and raising constructions;
2. Shared heads and dependents in coordination;
3. Co-reference in relative clause constructions;
4. Modifier specialization by means of case markers;
5. Null nodes for elided predicates.

2.1 Added subject relations

In the case of control and raising constructions, the subject of the subordinated non-finite clause is added. Consider the following examples, with controlled and raised subjects marked in bold:

1) Subject control: *La **mamma** ha promesso a Maria di comprare il pane* 'The **mother** promised Maria to buy the bread'
2) Object control: *La mamma ha convinto **Maria** a comprare il pane* 'The mother convinced **Maria** to buy the bread'
3) Oblique control: *La mamma ha chiesto a **Maria** di comprare il pane* 'The mother asked **Maria** to buy the bread'
4) Subject raising: *La **mamma** sembra apprezzare il pane integrale* 'The **mother** seems to like whole bread'

Figure 1 shows the UD representation of sentence 3), where the added subject relation (marked as `nsubj:xsubj`) is represented as an "enhanced arc" (in blue).

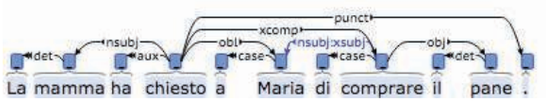
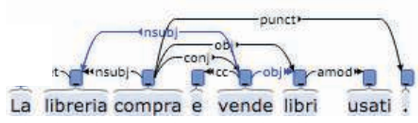


Figure 1. Enhanced representation of oblique control

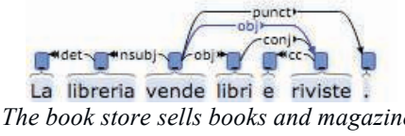
Control and raising predicates are superficially very similar, with a main difference: whereas Raising predicates have a 'non-thematic' argument, all arguments of Control predicates are 'thematic'. Such a distinction is neutralized in the enhanced UD representation. In both cases, however, the selection of the controlled/raised argument is lexically-driven.

2.2 Sharing in coordination

Coordination is another major source of potential enhancements, as information shared among conjuncts is typically attached only to the first conjunct and could be propagated to the other conjuncts, where this is applicable. In propagating information, it is useful to distinguish two cases, according to whether *dependents* of the first conjunct are propagated or the *head* of the first conjunct is propagated instead. Figure 2 shows Italian examples for each case.



The book store buys and sells used books.

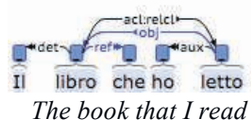


The book store sells books and magazines

Figure 2. a) Dependents propagation b) Head propagation

2.3 Co-reference in relative clauses

In basic UD, relative pronouns are normally attached to the main predicate of the relative clause, typically as nominal subjects (`nsubj`) or direct objects (`obj`). In the corresponding *enhanced* graph, the relative pronoun is linked to its antecedent with the `ref` relation and its dependency to the head of the relative clause is transferred to the antecedent itself, as exemplified in Figure 3 where it can be observed that the resulting enhanced representation contains a cycle.



The book that I read

Figure 3. Relative clauses

2.4 Specialization of relations

Adding case information to the relation name of non-core dependents serves the purpose of disambiguating their semantic role. This information is expressed in terms of the preposition or the subordinating conjunction introducing non-core dependents. In particular: `nmod` and `obl` relation labels, respectively marking nominal and oblique modifiers introduced by prepositions, are augmented with language specific case information; `acl` and `advcl` labels, corresponding respectively to noun modifying clauses and adverbial clauses, are augmented with markers introducing them. A similar type of specialization also applies to the `conj` dependency label linking conjuncts in coordinated structures, which is specialized with respect to the conjunction type (*e*, *o, oppure* …), as identified by the lemma of the `cc` dependency (i.e. the relation between a

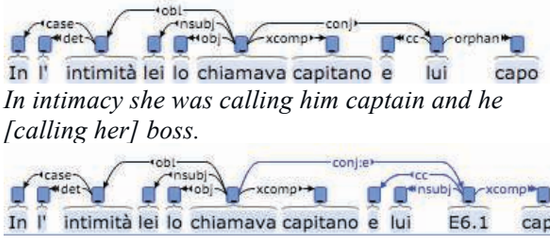
conjunct and a preceding coordinating conjunction).

After having dinner he went home

Figure 4. Adding case and mark information to labels

2.5 Null nodes for elided predicates

Special null nodes are added in clauses to stand for a predicate which is elided; other cases of ellipsis are not being dealt with in the current UD guidelines due to major difficulties in their reconstruction. This type of enhancement occurs when the basic (i.e. pre-enhancement) tree contains an orphan relation which in the enhanced graph is removed and replaced by the reconstructed explicit syntactic structure. A new null node is added in place of the missing predicate and dependencies are redirected. Figure 5 shows an example of predicate elision, along with the enhanced version which introduces a new node (labeled as E6.1) obtained as a copy of the token 'chiamava'.



In intimacy she was calling him captain and he [calling her] boss.

Figure 5. Null nodes for elided predicates

This is the most problematic among the foreseen UD enhancements, due to several reasons such as: correct insertion points are difficult to anticipate; phraseological verbs and verbs with clitics (either in pronominal form or with clitic complements, see example in Figure 5) would require copying a variable number of tokens (the verb and the object with a shift in gender in the case at hand), which is not always easy to be identified; the appropriate syntactic role of the dependents of the added (i.e. recovered) predicate must be inferred by proper alignment with the dependents of the originally explicit predicate. Moreover, the proposed UD treatment requires a major change in the treebank format with the addition of new tokens with special labeling and numbering. Therefore, the introduction of null nodes calls for an *ad hoc* treatment and introduces a complexity in the processing of the treebank which is not fully justified if the aim is only to address the cases of predicate elision, for the fact that this is a rare phenomenon in treebanks. Other cases of elision, such as subject elision, are much more meaningful for Italian.

2.6 Open issues

Besides the standard enhancements foreseen for UD illustrated above, we are currently evaluating cases that could be treated as such for Italian, and could possibly be relevant for other languages as well. These include:

- case information, which could also be added for some core relations such as `ccomp`. Consider as an example the following sentences: *Non so se verrà domani* 'I don't know whether (he) will come tomorrow' vs *Non so quando arriverà* 'I don't know when (he) will arrive'. Without enhancing the `ccomp` relation, the semantics of the subordinated clause (conditional vs temporal) remains underspecified;
- null nodes for elided subjects: Italian is a pro-drop language and the omission of explicit subjects occurs quite frequently in actual language usage; according to Bates (1976), the pro-drop rate by adults is 70%. The addition of null nodes for subject ellipsis could significantly enhance the syntactic representation with a view to information extraction tasks.

The typology of representation enhancements could also be further extended to neutralize diathesis alternations, as proposed by Candito *et al.* (2017) for French. In what follows, we focus on the standard UD enhancements, excluding the treatment of predicate elision for which more careful investigation and detailed guidelines are required.

Table 1. Guessing step: additional annotations

ExtraSubjOf=*id*	token *id* is head of a new arc to be added to current token
RefOf=*id*	
PropagateDepTo=*id*	
PropagateHeadWith=*label*	*label* is the string suggested to propagate or to specialize a relation
CaseSpec=*label*	
MarkSpec=*label*	
CcSpec=*label*	

3 Developing an enhanced UD gold treebank for Italian

UD enhanced representation cannot be generated through a completely automatic process: this is a task that entails a global vision of the tree to be completed and often requires additional linguistic knowledge concerning e.g. raising/control

properties and/or selectional preferences of predicates. To build the enhanced Italian UD Treebank (henceforth, e-IUDT), we followed a three-step approach, articulated as follows:

1. *Guessing*: by making use of heuristics, a script suggests target nodes whose representation might be enhanced, e.g. the best extra subject candidate(s) in raising/control constructions, or the heads/dependents to be propagated in coordinated constructions. During this step, additional annotations are produced in the representation of involved tokens. For example, the annotation ExtraSubjOf = *j* added to token *i* is an indication that *i* is an additional subject headed by *j*. In other cases, the additional annotation indicates a label to be used for specializing a given relation or whether a conjunct should be propagated. Table 1 summarizes the additional annotations used;
2. *Revising*: the human annotator is called to validate the proposed changes, automatically generated during the previous step;
3. *Enhancing*: validated additional annotations are used to automatically generate the enhanced UD representation. Enhancements are not limited to retyping or addition of dependencies; in some cases, they involve the reshaping of the dependency graph, and for this reason an automatic transformation reduces the chances of occasional errors.

The heuristics behind the guessing step make use of lexical resources extracted from the corpus itself: this is the case, for example, of lexical information on raising/control properties of predicates, guiding the identification of extra-subject candidates.

Following the three-step strategy sketched above, we built a gold standard e-IUDT resource on top of the development data set of the Italian UD treebank (Release 2.2), constituted by 11,908 tokens. In Table 2, the first two columns (headed by "IT DEV (GOLD)") summarize the enhancements contained in the developed resource, which involve 21,75% of the words. Most of them are represented by the specialization of modifiers and conjoining relations, immediately followed by head propagation, relative clauses and extra-subjects. Interestingly enough, it can be noticed that the distribution of enhancements remains quite similar across different subsets of the same language (e.g. the development vs test sets for Italian), whether manually revised (dev) or not (test), or for another language, English.

4 A language-independent rule-based UD enhancer

Different cross-lingual techniques have been developed for adding enhanced dependencies to existing UD treebanks, both rule-based (Schuster and Manning 2016) and data-driven (Nyblom *et al.*, 2013). The modularity of the approach proposed for e-IUDT construction created the prerequisites for reusing some of these components for implementing an UD enhancing module. In what follows, we report preliminary results achieved by transforming the heuristics of the *Guessing* module into language-independent ones. Instead of using language-specific lexical information on raising/control properties of verbs for identifying extra-subject candidates, following the general UD strategy we used the heuristic according to which the controlled / raised subject of the embedded clause follows the obliqueness hierarchy, i.e. it is the object of the next higher clause, if there is one, or else its subject. Such a strategy was extended to foresee also oblique complements as controlled / raised subjects. The output of the *Guessing* module is directly passed to the *Enhancing* component. In order to test effectiveness and generality of the approach we tested the rule-based language-independent enhancer on the Italian and English development sets, both available as gold datasets.

Table 2. Enhanced relations

	IT DEV (GOLD)		**IT TEST (SILVER)**		**EN DEV (GOLD)**		**EN TEST (GOLD)**	
words	11.908		10.417		25.150		17.658	
enhancements	2.590	21,75%	2.275	21,84%	4.255	16,92%	3.595	20,36%
xsubj	69	2,66%	69	3,03%	342	8.04%	251	6,98%
ref	127	4,90%	210	9,23%	111	2,61%	274	7,62%
conj specializations	322	12,4%	266	11,7%	810	19,03%	532	14,80%
dep propagation*	45	1,7%	36	1,6%	165	3,9%	103	2,87%
head propagation*	250	9,7%	230	10,1%	478	11,2%	413	11,49%
other specializations	1.777	68,6%	1.464	64,4%	2.349	55%	2.022	56,24%

For evaluation, we used an adaptation of the evaluation script used in the evaluation campaign EVALITA 2014 (Bosco *et al.*, 2014), which is based on a set of relations extracted from the enhanced graph and for each of them computes *Precision*, *Recall* and *F1*. The evaluation focused on enhanced relations, thus allowing to analyze the complexity of the task. Table 3 reports the results achieved with the following gold data sets: **IT-dev**, the development dataset from UD-ISDT 2.2, enhanced as described above; **EN-dev** and **EN-test**, the development and test English datasets from UD-EWT 2.2.

Table 3. Precision, recall and F1 for enhanced relations

	UAS			LAS		
	P	R	F1	P	R	F1
IT-dev	99,7	99,8	99,8	99,5	99,6	99,6
EN-dev	98,2	99,3	98,8	96,2	97,2	96,7
EN-test	99,2	99,0	99,0	97,8	97,6	97,6

Table 4. Recall and Precision for enhancement type

	IT-dev		EN-dev		EN-test	
	R	P	R	P	R	P
xsubj	92,7	98,4	100,0	99,4	99,6	99,0
ref	100,0	100,0	99,1	86,6	99,3	94,4
conj spec	99,7	100,0	98,2	94,9	97,9	97,6
other specs	99,9	100,0	97,0	96,7	98,2	98,1
propagation	97,8	95,7	97,1	97,3	95,5	98,2

For Italian, despite the de-lexicalization of the Guessing module, UAS and LAS results are quite high. Results are very high also when enhancement is carried out against different sets of the English UD Treebank. A qualitative error analysis was also performed. Table 4 details recall and precision achieved for the different types of enhancements, for both Italian and English.

The main sources of errors turned out to be:

- the identification of extra-subjects, performed on the basis of heuristics rather than lexical information. This is particularly true for Italian, for both P and R;
- the specialization of relations with case markers, which turned out to be particularly problematic for multi-word markers. This can be observed mainly for English, for which a different strategy is followed in their representation;
- dependent propagation in coordinated constructions, which is not always easy for both languages. For Italian, the interference with pro-drop subjects should also be considered;
- other problematic cases include non-homogenous conjuncts for which the propagation of dependents or heads cannot always be easily carried out.

An example follows where, without lexical information, the identification of extra subjects fails. Consider the sentence *I carri armati ... andavano a Budapest ... a spegnere i fuochi* 'The tanks ... went to Budapest ... to extinguish the fires'. In UD, the `obl` relation covers both lexically realized indirect objects and other oblique complements: however, without distinguishing between the two it is impossible to recover the extra subject of the infinitive clause. A suggestion could be to introduce a specialization of the `obl` relation for identifying indirect objects.

Dependency specialization turned out to be a challenging conversion case when applied to the English UD treebank: problems encountered were somehow unexpected, being mostly due to a different strategy for annotating multi-word case markers, not always compliant with the general UD annotation guidelines. This explains the lower results reported in Table 3 for English with respect to Italian.

5 Conclusions

We extended the Italian UD Treebank with an enhanced representation level: Italian is now among the few languages within UD with a gold enhanced Treebank which will be part of Release v2.3. The modules used to semi-automatically build e-IUDT were delexicalized to carry out cross-language enhancements: preliminary results for both Italian and English are promising. The contribution also includes better and more detailed specifications to the constantly in-progress guidelines. Current developments include: from a mono-lingual perspective, extension of the typology of enhancements; from the multi-lingual perspective, testing and extending the enhancement component successfully used with English for other languages.

References

Bates Elisabeth. 1976. *Language and context: The acquisition of pragmatics*. New York, NY: Academic Press.

Cristina Bosco, Vincenzo Lombardo, Leonardo Lesmo, Daniela Vassallo. 2000. Building a treebank for Italian: a data-driven annotation schema. In Proceedings of LREC 2000, Athens, Greece.

Cristina Bosco, Simonetta Montemagni, Maria Simi. 2012. Harmonization and Merging of two Italian

Dependency Treebanks, Workshop on Merging of Language Resources, in Proceedings of LREC 2012, Workshop on Language Resource Merging, Instanbul, May 2012, ELRA, pp. 23–30.

Cristina Bosco, Simonetta Montemagni, Maria Simi. 2013. Converting Italian Treebanks: Towards an Italian Stanford Dependency Treebank. In: *ACL Linguistic Annotation Workshop & Interoperability with Discourse*, Sofia, Bulgaria.

Cristina Bosco, Felice Dell'Orletta, Simonetta Montemagni, Manuela Sanguinetti, Maria Simi. 2014. The Evalita 2014 Dependency Parsing task, CLiC-it 2014 and EVALITA 2014 Proceedings, Pisa University Press, ISBN/EAN: 978-886741-472-7, 1–8.

Marie Candito, Bruno Guillaume, Guy Perrier, Djamé Seddah. 2017. Enhanced UD Dependencies with Neutralized Diathesis Alternation, *Depling 2017 - Fourth International Conference on Dependency Linguistics*, Sep 2017, Pisa, Italy. 2017

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The Stanford typed dependencies representation. In COLING Workshop on Cross-framework and Cross-domain Parser Evaluation.

Marie-Catherine de Marneffe, Miriam Connor, Natalia Silveira, Bowman S. R., Timothy Dozat, Christopher D. Manning. 2013. More constructions, more genres: Extending Stanford Dependencies, Proc. of the Second International Conference on Dependency Linguistics (DepLing 2013), Prague, August 27–30, Charles University in Prague, Matfyzpress, Prague, pp. 187–196.

Marie-Catherine de Marneffe and Christopher D. Manning. 2013. Stanford typed dependencies manual, September 2008, Revised for the Stanford Parser v. 3.3 in December 2013.

Marie-Catherine De Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, Christopher D. Manning. 2014. Universal Stanford Dependencies: a Cross-Linguistic Typology. In: *Proc. LREC 2014*, Reykjavik, Iceland, ELRA.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, Daniel Zeman. 2016. Universal Dependencies v1: A Multilingual Treebank Collection. In Proceedings of LREC.

Simonetta Montemagni, Maria Simi. 2007. The Italian dependency annotated corpus developed for the CoNLL–2007 shared task. Technical report, ILC–CNR.

Jenna Nyblom, Samuel Kohonen, Katri Haverinen, Tapio Salakoski and Filip Ginter. 2013. Predicting conjunct propagation and other extended Stanford Dependencies. Proceedings of the Second International Conference on Dependency Linguistics (DepLing 2013), pp 252–261, Prague, August 27–30.

Maria Simi, Cristina Bosco, Simonetta Montemagni. 2008. Less is More? Towards a Reduced Inventory of Categories for Training a Parser for the Italian Stanford Dependencies. In: *Proc. LREC 2014*, 26–31, May, Reykjavik, Iceland, ELRA.

Schuster, Sebastian and Christopher D. Manning. Enhanced English Universal Dependencies: An Improved Representation for Natural Language Understanding Tasks." LREC (2016).

Analysing the Evolution of Students' Writing Skills and the Impact of Neo-standard Italian with the help of Computational Linguistics

Rachele Sprugnoli FBK, Trento sprugnoli@fbk.eu
Sara Tonelli FBK, Trento satonelli@fbk.eu
Alessio Palmero Aprosio FBK, Trento aprosio@fbk.eu
Giovanni Moretti FBK, Trento moretti@fbk.eu

Abstract

English. We present a project aimed at studying the evolution of students' writing skills in a temporal span of 15 years (from 2001 to 2016), analysing in particular the impact of neo-standard Italian. More than 2,500 essays have been transcribed and annotated by teachers according to 28 different linguistic traits. We present here the annotation process together with the first data analysis supported by NLP tools.

Italiano. *In questo contributo presentiamo un progetto finalizzato allo studio dell'evoluzione delle abilità di scrittura negli studenti in un arco temporale di 15 anni (dal 2001 al 2016), e in particolare all'analisi dell'impatto dell'italiano neo-standard. In questo contesto, più di 2.500 temi sono stati trascritti e annotati da insegnanti, registrando la presenza di 28 diversi tratti linguistici. Il presente studio illustra il processo di annotazione e le prime analisi dei dati con il supporto di strumenti TAL.*

1 Introduction

In this work, we present an extensive study on the evolution of high-school students' writing skills, taking into account essays spanning 15 years (from 2001 to 2016). In particular, we are interested in tracking the presence of expressions and constructions typical of neo-standard Italian (Berruto, 2012), in the light of the recent public discussion on the 'decline of Italian in schools' [1].

[1] See the open letter signed by around 600 University professors at http://gruppodifirenze.blogspot.it/2017/02/contro-il-declino-dellitaliano-scuola.html.

The Italian neo-standard is the current linguistic register in Italy, in which forms previously considered colloquial have become widely accepted in the national language.

We analyse more than 2,500 essays written by students from different high-schools in the Autonomous Province of Trento during the exit exam (the so-called *Maturità*). The study is the outcome of a project comprising different steps: *i)* digital acquisition and transcription of thousands of essays balancing their distribution across school years and school types; *ii)* computer-assisted annotation of some linguistic traits of interest; *iii)* diachronic analysis of the traits. While the first step has been carried out by the Istituto provinciale per la Ricerca e la Sperimentazione educativa (IPRASE), we led steps *ii)* and *iii)*, which are discussed in the next sections. Beside an in-depth and diachronic study of the evolution of students' writing skills, a major contribution of this paper is also the release of the corpus in the form of embeddings and n-grams.

2 Corpus Collection

The staff of IPRASE have digitized and transcribed essays stored in the archives of 21 secondary schools located in different areas of Trentino Province. These areas include both the two major cities, Trento and Rovereto, but also other communities in the valleys (Val di Fiemme, Val di Non, Valsugana) and Riva del Garda. Nine different types of schools were involved: liceo classico, liceo scientifico, liceo artistico, liceo linguistico, liceo musicale e coreutico, liceo delle scienze umane, istituto tecnico tecnologico, istituto tecnico economico and istituto professionale. Six school years were chosen between 2000-2001 and 2015-2016, thus having a temporal span of 15 years for a total of 2,544 essays and almost 1.5 million words. Table 2 shows the distribution of essays per year with the corresponding number of

words. These essays are of the so-called type B, that requires students to write a short essay or a newspaper article. Students can choose between 4 areas: artistic-literary, socio-economic, technical-scientific, historical-political. For each area, a title is given together with a set of reference materials. For example, students writing an essay of type B with historical-political content in 2014 were asked to comment some excerpts from Hannah Arendt, Ghandi and Martin Luther King about violence and non-violence in the XX Century.

SCHOOL YEAR	#ESSAYS	#WORDS
2000-2001	417	244,312
2003-2004	439	270,388
2006-2007	430	258,188
2009-2010	429	245,821
2012-2013	421	234,329
2015-2016	408	224,776
TOTAL	2,544	1,477,814

Table 1: Number of essays and words per school year in our corpus.

Due to privacy reasons, we are not allowed to distribute the full texts of the corpus. However, we release both word vectors and n-grams of the essays. We build three types of embeddings with 300 dimensions: the GloVe embeddings based on linear bag-of-words contexts (Pennington et al., 2014), Levy and Goldberg's ones using dependency parse-trees (Levy and Goldberg, 2014), and fastText embeddings with bag of character n-grams (Bojanowski et al., 2017). As for the n-grams, we generated both case-sensitive and case-insensitive sequences per school year, considering the range [1,5]. N-grams and pre-trained word embeddings in text format are available for download on our website[2]. In addition, word vectors are visualized through a dedicated stand-alone version of the TensorFlow embedding projector (Smilkov et al., 2016)[3].

3 Description of Linguistic Traits

Around 20 teachers have been involved in the annotation of essays using the CAT platform (Bartalesi Lenzi et al., 2012), through which they had to annotate between 100 and 150 essays each. We also organised 2 preliminary training sessions with the teachers to show the tool functionalties, explain the annotation process and make sure that everyone followed the guidelines[4]. Note that the teachers knew neither the name of the student writing the essay nor his/her school. Moreover, for all of them, it was the first time using an electronic platform for text annotation.

We briefly present in Table 2 the traits that the teachers had to mark on each essay. The goal of the annotation is to detect the presence of linguistic traits that were deemed relevant to diachronically study style and complexity evolution by IPRASE experts and teachers. This approach is therefore rather different from the standard essay correction that is usually performed by teachers, and for this reason the training phase was particularly relevant.

The list of traits to include in the project was mainly inspired by the work of (D'Achille, 2003) and (Boscolo and Zuin, 2015). The goal of this annotation was to cover all levels of linguistic analysis, including lexical choices (e.g. trait 8 and 20), grammar (e.g. trait 1 and 2), semantics (e.g. trait 15) and discourse structure (e.g. trait 24 and 25).

In the first Table column, we mark traits that were identified in a fully automatic way (A), those that were annotated semi-automatically (S), and the manual ones (M). For those marked with S, we pre-processed the essays using the Tint NLP tool (Aprosio and Moretti, 2018) enriched with a set of new modules developed to add all information needed to speed up annotation. For example, for traits 21 and 23 we matched the essay n-grams with pre-defined lists of politically correct expressions and cliché expressions provided by IPRASE, so that teachers could see in the CAT interface the corresponding markables already highlighted, and they just had to validate them. For other traits, for example 10 and 11, they had to add attributes to the markables. For some traits, we performed pre-annotation using available external resources, for example the list of affixes included in the *derIvaTario*[5] for trait 13 (Talamo et al., 2016).

After the initial training phase, the average annotation time for each essay through the web interface was 30 minutes. We roughly estimate that the same task would take at least one hour on a standard Word document. Another advantage of using

[2] https://dh.fbk.eu/technologies/students-essays

[3] http://dhlab.fbk.eu/TemiVectors/

[4] A complete version of the annotation guidelines (in Italian) is available at this link: http://bit.do/erd9P

[5] http://derivatario.sns.it/

Type	ID	Trait	Description
S	1	Monosyllables	Annotate monosyllabic terms with a wrong accent
A	2	Apostrohpes	Annotate the wrong use of apostrophes for the article 'un'
S	3	Capitalized words	Annotate wrong capitalisations inside a sentence
A	4	"il"	Annotate the wrong use of "il"
S	5	Personal pronouns	Annotate personal pronouns and mark when 'loro' is used to mean 'a loro'
S	6	"Gli"	Annotate different uses of 'gli' including mistakes
S	7	"Questo"	Annotate when 'quest*' is used to refer generically to the discourse context
A	8	Generic words	Annotate generic words such as 'bello', 'brutto', 'fare', 'dire', 'cosa'
S	9	Indicativo imperfetto	Annotate different types of imperfetto (e.g. in place of conjunctive, in hypothetical clauses)
S	10	Gerund	Annotate different types of gerundio
S	11	Indicativo presente	Annotate different types of indicativo presente
A	12	'stare / andare'	Annotate when 'stare' / 'andare' are used properly or in phrasal constructions
S	13	Affixes	Annotate words created using specific affixes such as -anti, '-dopo', '-trans', '-ismo', '-izzare', ...
S	14	Number of words, clauses, sentences	Count the number of words, clauses and sentences. Annotate verbless clauses when not in the title
S	15	Connectives 1	Annotate the use of very generic connectives ('che / dove / allora') and their correct or improper use
S	16	Connectives 2	Count complex connectives such as 'nondimeno', 'sebbene', 'qualora' and annotate their use
S	17	Punctuation	Count punctuation marks: [; : ! " ... , .] and annotate their correct or improper use
S	18	Connectives beginning a sentence	Identify connectives such as 'perché' and 'quando' at the beginning of a sentence and annotate their use
S	19	Informal register	Annotate a set of expressions belonging to an informal register ('della serie', 'tipo', 'troppo forte', etc.)
S	20	Anglicisms	Annotate adapted and not adapted anglicisms
S	21	Politically correct terms	Annotate politically correct terms such as 'ministra', 'sindaca', 'non vedente', etc.
S	22	Multiwords	Annotate multiword expressions (*polirematiche*)
S	23	Cliché expressions	Annotate cliché expressions from a predefined list
M	24	Dislocated clauses	Annotate left or right dislocated sentences
S	25	Cleft sentences	Annotate cleft sentences
S	26	'li'	Annotate 'li' when it is mistakenly used instead of 'gli'
A	27	Euphonic 'd'	Annotate when 'd' is added before a word starting with a vowel
M	28	Other traits	Add other relevant linguistic phenomena that are not captured by previous traits

Table 2: List of annotated traits with a label for Automatic (A), Semi-automatic (S) or Manual (M)

the CAT interface was the possibility to have all annotations in a consistent format, easily export them to compute statistics and make comparisons.

4 Linguistic Analysis

We present here an analysis of some traits of interest. We focus in particular on traits that are, at least in part, automatically annotated and counted (marked with A or S in Table 2), because the work of those requiring a manual annotation is still in progress. For each trait we compute the observed relative frequency per 10,000 words. This normalization has allowed us to have more easily comparable and legible numbers. Furthermore, we calculate the Gulpease index to monitor writing complexity (Lucisano and Piemontese, 1988). This score has been specifically defined for measuring the readability of Italian texts based on proficiency level and it combines two linguistic variables: the average length of the words and of the sentences in a document. Its value determines the level of readability of a text: the higher the score, the easier the text is to understand.

To extract reliable measures of students' language use, we removed from the texts the quotations present in the essays citing the reference material provided together with the topic. This preprocessing step was performed by adopting the FuzzyWuzzy package[6], a Java fuzzy string matching implementation, and the Stanford CoreNLP quote annotator[7]. These tools allow us to recognize text reuse both when it is explicitly signaled by quotes and when there is no overt signal. The average percentage of quotations within the corpus is 1.9% but it varies a lot among the essays, reaching up to 46% of the content in some cases. The following is an example taken from an essay about the pursuit of happiness in 2010 for the socio-economic area. The snippet in bold, containing one of the complex connectives of trait 16, was automatically removed: *La riflessione di Zygmunt Bauman sembra essere una risposta:* ***"L'incertezza è l'habitat naturale della vita umana, sebbene la speranza di sfuggire ad essa sia il motore delle attività umane."***

After removing quotations, we obtain the following results for the automatically annotated traits:

Trait 8 - Generic Words. We trace the presence of semantically generic and polysemic words, which are frequently used in neo-standard Italian (Fig. 1). In particular, lemmas 'fare', 'dire', and 'cosa' (*to make*, *to say*, *thing*) show a decrease in occurrence in the last two school

[6] https://github.com/xdrop/fuzzywuzzy

[7] https://stanfordnlp.github.io/CoreNLP/quote.html

years considered (2012-2013 and 2015-2016). For example, the relative incidence of 'fare' every 10,000 tokens goes from 42.013 in 2000-2001 to 26.857 in 2015-2016 indicating an effort to use more specific and differentiated expressions. Liceo classico has the lowest ratio for 'fare' and 'dire', whereas istituto professionale has an occurrence above the average for 'fare' and 'cosa'.

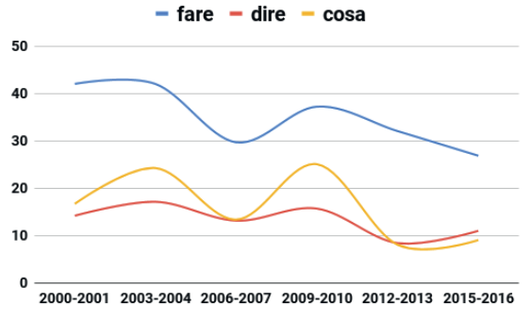


Figure 1: Observed relative frequency of three generic words per 10,000 tokens.

Trait 14 - Nominal Sentences. Sentences without a verbal predicate are a typical feature of news style and juvenile writing, to make the text dramatic and concise (Dardano, 1986; Ardrizzo and Gambarara, 2003). This tendency is present also in our corpus with an impact of 6.1% over the total amount of sentences, after removing the title of the essays. The trait is particularly relevant in liceo classico with an above-average percentage of 7.7%.

Trait 16 - Complex Connectives. The lack of complex connectives is another indicator of neo-standard Italian. As shown in Figure 2, 'nondimeno' is never used by students and also 'qualora' and 'giacché', used mostly in liceo classico, disappear in the last two school years from all the essays. 'Affinché' is adopted in all school types with the only exception of liceo artistico, in which complex connectives are barely used.

Trait 17 - Punctuation. Over the last two school years considered in our analysis, there has been an overall decline in the use of punctuation with the exception of question marks (see Figure 3). The frequent use of question marks is inherited from the style of news (Buroni, 2009); however, the peak in 2009-2010 is also due to the presence of a question in the title of an essay (*Siamo soli?*), which led students use the same rhetorical device in their texts. The presence of punctuation not suitable for medium-high style such as multiple exclamation marks and suspension points is also decreasing.

Trait 27 - Euphonic 'd'. Following a recent grammatical rule[8], the euphonic 'd' should be introduced only when the conjunction 'e' or the preposition 'a' are followed by a word starting with the same vowel: e.g., *ed ecco*, *ad andare*. However, this rule is not followed in the essays and the presence of 'd' between two different vowels is higher than the one between the same vowels (33.8 versus 17.6 of relative frequency). Besides, while the disappearance of this trait is considered a characteristic of neo-standard Italian (D'Achille, 2003), this trend is not found in our corpus, where the relative frequency of euphonic 'd' is only 6 points lower than the same conjunction without 'd' preceding a vowel.

Gulpease. We computed the Gulpease index to see whether there has been a decrease of complexity, i.e. an increase in readability, over time. Contrary to our expectations, the average readability of essays has slightly decreased in the last two years considered, with a drop of 1.8 points, bringing it below 50. This corresponds to texts that are quite difficult to read for a person with a medium school degree (*diploma di scuola media* in the Italian school system), but not too challenging for a person with a high school degree. Moreover, values do not change much across different school types.

These preliminary analyses show that the impact of neo-standard Italian is multi-faceted and, while some traits confirm that students' language is getting simpler and less formal (e.g. overall decline of punctuation), some others seem to contradict this finding (e.g. decline in the use of 'fare', 'dire', 'cosa'). Also the differences across school types are not clear-cut and consistent.

5 Related Work

While several works in the past have focused on the creation and analysis of corpora to study students' mistakes, their writing quality and their rate of progress over the year (Parr, 2010; McNamara et al., 2010), they have mainly dealt with English essays. A notable exception are two corpora in

[8] http://www.accademiadellacrusca.it/it/lingua-italiana/consulenza-linguistica/domande-risposte/d-eufonica

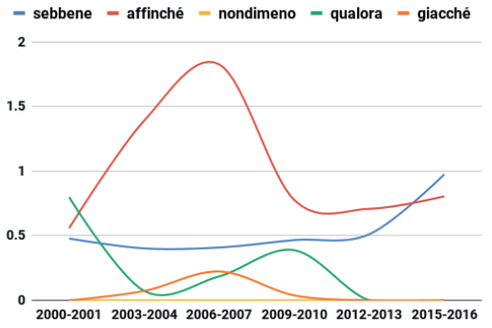


Figure 2: Observed relative frequency of complex connectives per 10,000 words.

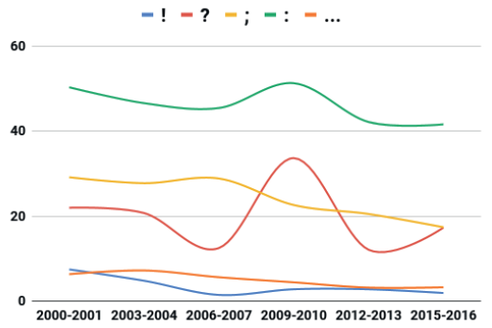


Figure 3: Observed relative frequency of punctuation per 10,000 words.

German, the KoKo corpus of argumentative essays to study pupils' writing competences (Abel et al., 2016) and the corpus collected by Berkling et al. (2014) to study different error categories.

As for Italian, a relatively small number of studies has been carried out with various goals. The projects TIscrivo (2011-2014) and TIscrivo 2.0 (2014-2017)[9] have been launched to investigate the writing skills of primary schools and lower secondary schools in Southern Switzerland (Cignetti et al., 2016), and have led to the creation of a corpus of 1,735 essays. Another research deals with the analysis of oral and written productions of Italian children in primary schools, and 200 texts have been collected in the ISACCO corpus (Brunato and dell'Orletta, 2015). Another corpus, called CItA (Barbagli et al., 2016), includes texts written in the first and second year of lower secondary school, tracking L1 writing competence of the same group of students over two school years.

Compared to previous works, our analysis is different in several ways. First, none of the previous studies considers a text span of 15 years. Then, the traits to be annotated are different: we do not focus on mistakes, but on indicators of neostandard Italian. Finally, our interest lies also in the annotation workflow, studying how NLP can support the identification of such traits and implementing the necessary processing modules to speed up annotation.

[9] http://dfa-blog.supsi.ch/tiscrivo/

6 Conclusions

In this work, we have presented a project aimed at tracking the evolution of students' writing skills over time. The goal of this work was not only to introduce the corpus collection and annotation activities, but also to show how this kind of projects can benefit from NLP by speeding up annotation and increasing data consistency. In the future we will complete the analysis of all the traits for a more comprehensive view of the role of neostandard Italian in students' essays. We will also use some of the manual annotations to train new NLP modules performing the same task automatically.

Acknowledgments

We would like to thank Chiara Motter from IPRASE for coordinating the corpus transcription, and the high-school teachers for annotating the essays.

References

Andrea Abel, Aivars Glaznieks, Lionel Nicolas, and Egon Stemle. 2016. An extended version of the koko german L1 learner corpus. In *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.*

Alessio Palmero Aprosio and Giovanni Moretti. 2018. Tint 2.0: An all-inclusive Suite for NLP in Italian. In *Proceedings of CLIC-it*.

Giuseppe Ardrizzo and Daniele Gambarara. 2003. *La comunicazione giovane*. Rubbettino Editore.

Alessia Barbagli, Pietro Lucisano, 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 2016, Portorož, Slovenia, May 23-28, 2016.*

Valentina Bartalesi Lenzi, Giovanni Moretti, and Rachele Sprugnoli. 2012. CAT: the CELCT Annotation Tool. In *In Proceedings of LREC 2012*, pages 333–338.

Kay Berkling, Johanna Fay, Masood Ghayoomi, Katrin Hein, Rémi Lavalley, Ludwig Linhuber, and Sebastian Stüker. 2014. A database of freely written texts of german school students for the purpose of automatic spelling error classification. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland.

Gateano Berruto. 2012. *Sociolinguistica dell'italiano contemporaneo*. Carocci.

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.

Pietro Boscolo and Elvira Zuin, editors. 2015. *Come scrivono gli adolescenti. Un'indagine sulla scrittura scolastica e sulla didattica della scrittura*. Il Mulino.

Dominique Brunato and Felice dell'Orletta. 2015. ISACCO: a corpus for investigating spoken and written language development in Italian school–age children. In *Proceedings of CLIC-it*.

Edoardo Buroni. 2009. Politicamente corretto? Aspetti grammaticali nei quotidiani politici della "Seconda Repubblica" tra norma, uso medio e finalità pragmatiche. *Studi di Grammatica Italiana*, 2007:107–163.

Luca Cignetti, Silvia Demartini, and Simone Fornara. 2016. *Come TIscrivo? La scrittura a scuola tra teoria e didattica*. Aracne.

Paolo D'Achille. 2003. *L'italiano contemporaneo*. Il mulino Bologna.

Maurizio Dardano. 1986. *Il linguaggio dei giornali italiani*, volume 18. Laterza.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *ACL (2)*, pages 302–308.

Pietro Lucisano and Maria Emanuela Piemontese. 1988. GULPEASE: una formula per la predizione della difficoltà dei testi in lingua italiana. *Scuola e città*, 3(31):110–124.

Danielle S. McNamara, Scott A. Crossley, and Philip M. McCarthy. 2010. Linguistic features of writing quality. *Written Communication*, 27(1):57–86.

Judy M. Parr. 2010. A dual purpose data base for research and diagnostic assessment of student writing. *Journal of Writing Research*, vol. 2(issue 2):129–150. Query date: 2018-06-25.

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.

Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B Viégas, and Martin Wattenberg. 2016. Embedding Projector: Interactive visualization and interpretation of embeddings. In *30th Conference on Neural Information Processing Systems (NIPS 2016)*.

Luigi Talamo, Chiara Celata, and Pier Marco Bertinetto. 2016. DerIvaTario: An annotated lexicon of Italian derivatives. *Word Structure*, 9(1):72–102.

Arretium or *Arezzo*? A Neural Approach to the Identification of Place Names in Historical Texts

Rachele Sprugnoli
Fondazione Bruno Kessler, Via Sommarive 18, Povo (TN)
sprugnoli@fbk.eu

Abstract

English. This paper presents the application of a neural architecture to the identification of place names in English historical texts. We test the impact of different word embeddings and we compare the results to the ones obtained with the Stanford NER module of CoreNLP before and after the retraining using a novel corpus of manually annotated historical travel writings.

Italiano. *Questo articolo presenta l'applicazione di un'architettura neurale all'identificazione dei nomi propri di luogo all'interno di testi storici in lingua inglese. Abbiamo valutato l'impatto di vari word embedding e confrontato i risultati con quelli ottenuti usando il modulo NER di Stanford CoreNLP prima e dopo averlo riaddestrato usando un nuovo corpus di lettaratura di viaggio storica manualmente annotato.*

1 Introduction

Named Entity Recognition (NER), that is the automatic identification and classification of proper names in texts, is one of the main tasks of Natural Language Processing (NLP), having a long tradition started in 1996 with the first major event dedicated to it, i.e. the Sixth Message Understanding Conference (MUC-6) (Grishman and Sundheim, 1996). In the field of Digital Humanities (DH), NER is considered as one of the important challenges to tackle for the processing of large cultural datasets (Kaplan, 2015). The language variety of historical texts is however greatly different from the one of the contemporary texts NER systems are usually developed to annotate, thus an adaptation of current systems is needed.

In this paper, we focus on the identification of place names, a specific sub-task that in DH is envisaged as the first step towards the complete geoparsing of historical texts, which final aim is to discover and analyse spatial patterns in various fields, from environmental history to literary studies, from historical demography to archaeology (Gregory et al., 2015). More specifically, we propose a neural approach applied to a new manually annotated corpus of historical travel writings. In our experiments we test the performance of different pre-trained word embeddings, including a set of word vectors we created starting from historical texts. Resources employed in the experiments are publicly released together with the model that achieved the best results in our task[1].

2 Related Work

Different domains - such as Chemistry, Biomedicine and Public Administration (Eltyeb and Salim, 2014; Habibi et al., 2017; Passaro et al., 2017) - have dealt with the NER task by developing domain-specific guidelines and automatic systems based on both machine learning and deep learning algorithms (Nadeau and Sekine, 2007; Ma and Hovy, 2016). In the field of Digital Humanities, applications have been proposed for the domains of Literature, History and Cultural Heritage (Borin et al., 2007; Van Hooland et al., 2013; Sprugnoli et al., 2016). In particular, the computational treatment of historical newspapers has received much attention being, at the moment, the most investigated text genre (Jones and Crane, 2006; Neudecker et al., 2014; Mac Kim and Cassidy, 2015; Neudecker, 2016; Rochat et al., 2016).

`Person`, `Organization` and `Location` are the three basic types adopted by general-purpose NER systems, even if different entity types can be detected as well, depending on

[1] https://dh.fbk.eu/technologies/place-names-historical-travel-writings

the guidelines followed for the manual annotation of the training data (Tjong Kim Sang and De Meulder, 2003; Doddington et al., 2004). For example, political, geographical and functional locations can be merged in a unique type or identified by different types: in any case, their detection has assumed a particular importance in the context of the spatial humanities framework, that puts the geographical analysis at the center of humanities research (Bodenhamer, 2012). However, in this domain, the lack of pre-processing tools, linguistic resources, knowledge-bases and gazetteers is considered as a major limitation to the development of NER systems with a good accuracy (Ehrmann et al., 2016).

Compared to previous works, our study focuses on a text genre not much investigated in NLP but of great importance from the historical and cultural point of view: travel writings are indeed a source of information for many research areas and are also the most representative type of intercultural narrative (Burke, 1997; Beaven, 2007). In addition, we face the problem of poor resource coverage by releasing new historical word vectors and testing an architecture that does not require any manual feature selection, and thus neither text pre-processing nor gazetteers.

3 Manual Annotation

We manually annotated a corpus of 100,000 tokens divided in 38 texts taken from a collection of English travel writings (both travel reports and guidebooks) about Italy published in the second half of the XIX century and the '30s of the XX century (Sprugnoli, 2018). The tag `Location` was used to mark all named entities (including nicknames like *city on the seven hill*) referring to:

- geographical locations: landmasses (*Janiculum Hill*, *Vesuvius*), body of waters (*Tiber*, *Mediterranean Sea*), celestial bodies (*Mars*), natural areas (*Campagna Romana*, *Sorrentine Peninsula*);
- political locations: areas defined by socio-political groups, such as cities (*Venice*, *Palermo*), regions (*Tuscany*, *Lazio*), kingdoms (*Regno delle due Sicilie*), nations (*Italy*, *Vatican*);
- functional locations: areas and places that serve a particular purpose, such as facilities (*Hotel Riposo*, *Church of St. Severo*), monuments and archaeological sites (*Forum Romanum*) and streets (*Via dell'Indipendenza*).

The three aforementioned definitions correspond to three entity types of the ACE guidelines, i.e., `GPE` (geo-political entities), `LOC` (locations) and `FAC` (facilities): we extended this latter type to cover material cultural assets, that is the built cultural inheritance made of buildings, sites, monuments that constitute relevant locations in the travel domain.

The annotation required 3 person/days of work and, at the end, 2,228 proper names of locations were identified in the corpus, among which 657 were multi-token (29.5%). The inter-annotator agreement, calculated on a subset of 3,200 tokens, achieved a Cohen's kappa coefficient of 0.93 (Cohen, 1960), in line with previous results on named entities annotation in historical texts (Ehrmann et al., 2016).

The annotation highlighted the presence of specific phenomena characterising place names in historical travel writings. First of all, the same place can be recorded with variations in spelling across different texts but also in the same text: for example, modern names can appear together with the corresponding ancient names (*Trapani gradually assumes the form that gave it its Greek name of Drepanum*) and places can be addressed by using both the English name and the original one, the latter occurring in particular in code-mixing passages (Sprugnoli et al., 2017) such as in: (*Byron himself hated the recollection of his life in Venice, and I am sure no one else need like it. But he is become a cosa di Venezia, and you cannot pass his palace without having it pointed out to you by the gondoliers.*). Second, some names are written with the original Latin alphabet graphemes, such as *Ætna* and *Tropæa Marii*. Then, there are names having a wrong spelling: e.g., *Cammaiore* instead of *Camaiore* and *Momio* instead of *Mommio*. In addition, there are several long multi-token proper names, especially in case of churches and other historical sites, e.g. *House of the Tragic Poet*, *Church of San Pietro in Vincoli*, but also abbreviated names used to anonymise personal addresses, e.g. *Hotel B.*. Travel writings included in the corpus are about cities and regions of throughout Italy thus there is a high diversity in the mentioned locations, from valleys in the Alps (*Val Buona*) to small villages in Sicily (*Capo S. Vito*). However, even if the main topic of the corpus is the descrip-

tion of travels in Italy, there are also references to places outside the country, typically used to make comparisons (*Piedmont, in Italy, is nothing at all like neighbouring Dauphiné or Savoie*).

4 Experiments

Experiments for the automatic identification of place names were carried out using the annotated corpus described in the previous Section. The corpus, in BIO format, was divided in a training, a test and a development set following a 80/10/10 split. For the classification, we tested two approaches: we retrained the NER module of Stanford CoreNLP with our in-domain annotated corpus and we used a BiLSTM implementation evaluating the impact of different word embeddings, including three new historical pre-trained word vectors.

4.1 Retraining of Stanford NER Module

The NER system integrated in Stanford CoreNLP is an implementation of Conditional Random Field (CRF) sequence models (Finkel et al., 2005) trained on a corpus made by several datasets (CONLL, MUC-6, MUC-7, ACE) for a total of more than one million tokens[2]. The model distributed with the CoreNLP distribution is therefore based on contemporary texts, most of them of the news genre but also weblogs, newsgroup messages and broadcast conversations. We evaluated this model (belonging to the 3.8.0 release of CoreNLP) on our test set and then we trained a new CRF model using our training data.

4.2 Neural Approach

We adopted an implementation of BiLSTM-CRF developed from the Ubiquitous Knowledge Processing Lab (Technische Universität Darmstadt)[3]. This architecture exploits casing information, character embeddings and word embeddings; no feature engineering is required (Reimers and Gurevych, 2017a). We chose this implementation because the authors propose recommended hyperparameter configurations for several sequence labelling tasks, including NER, that we took as a reference for our own experiments. More specifically, the setup suggested by Reimers and Gurevych (2017a) for the NER task is summarised below:

- dropout: 0.25, 0.25
- classifier: CRF
- LSTM-Size: 100
- optimizer: NADAM
- word embeddings: GloVe Common Crawl 840B
- character embeddings: CNN
- miniBatchSize: 32

Starting from this configuration, we evaluated the performance of the NER classifier trying different pre-trained word embeddings. Given that the score of a single run is not significant due to the different results producing by different seed values (Reimers and Gurevych, 2017b), we run the system three times and we calculated the average of the test score corresponding to the epoch with the highest result on the development test. We used Keras version 1.0[4] and Theano 1.0.0[5] as backend; we stopped after 10 epochs in case of no improvements on the development set.

4.2.1 Pre-trained Word Embeddings

We tested a set of word vectors available online, all with 300 dimensions, built on corpora of contemporary texts and widely adopted in several NLP tasks, namely: (i) GloVe embeddings, trained on a corpus of 840 billion tokens taken from Common Crawl data (Pennington et al., 2014); (ii) Levy and Goldberg embeddings, produced from the English Wikipedia with a dependency-based approach (Levy and Goldberg, 2014); (iii) fastText embeddings, trained on the English Wikipedia using sub-word information (Bojanowski et al., 2017). By taking into consideration these pre-trained embeddings, we cover different types of word representation: GloVe is based on linear bag-of-words contexts, Levy on dependency parse-trees, and fastText on a bag of character n-grams.

In addition, we employed word vectors we developed using GloVe, fastText and Levy and Goldberg's algorithms on a a subset of the Corpus of Historical American English (COHA) (Davies, 2012) made of more than 198 million words. The chosen subset contains more than 3,800 texts belonging to four genres (i.e., fiction, non-fiction, newspaper, magazine) published in the same temporal span of our corpus of travel writings. These

[2] https://nlp.stanford.edu/software/CRF-NER.html

[3] https://github.com/UKPLab/emnlp2017-bilstm-cnn-crf

[4] https://keras.io/

[5] http://deeplearning.net/software/theano/

historical embeddings, named HistoGlove, HistoFast and HistoLevy, are available online[6].

5 Results and Discussion

Table 1 shows the results of our experiments in terms of precision (P), recall (R) and F-measure (F1): the score obtained with the Stanford NER module before and after the retraining is compared with the one achieved with the deep learning architecture and different pre-trained word embeddings.

The neural approach performs remarkably better than the CFR sequence models with a difference ranging from 11 to 14 points in terms of F1, depending on the word vectors used. The original Stanford module produces much unbalanced results with the lowest recall and F1 but a precision above 82. In all the other experiments, scores are more balanced even if in the majority of the neural experiments recall is slightly higher than precision, meaning that BiLSTM is more able to generalise the observations of named entities from the training data. Although the training data are few, compared to the corpora used for the original Stanford NER module, they produce an improvement of 13.1 and 5.9 points on recall and F1 respectively, demonstrating the positive impact of having in-domain annotated data.

As for word vectors, dependency-based embeddings are not the best word representation for the NER task having the lowest F1 among the experiments with the neural architecture. It is worth noticing that GloVe, suggested as the best word vectors by Reimers and Gurevych (2017a) for the NER task on contemporary texts, does not achieve the best scores on our historical corpus. Linear bag-of-words contexts is however confirmed as the most appropriate word representation for the identification of Named Entities, given that HistoGloVe produces the highest scores for all the three metrics.

The improvement obtained with the neural approach combined with historical word vectors and in-domain training data is evident when looking in details at the results over the three files constituting the test set. These texts were extracted from two travel reports, "A Little Pilgrimage in Italy" (1911) and "Naples Riviera" (1907) and one guidebook, "Rome" (1905). The text taken from the latter book is particularly challenging for the presence of many Latin place names and locations related to the ancient (and even mythological) history of the city of Rome, e.g. *Grotto of Lupercus*, *Alba Longa*. As displayed in Table 2, Neural HistoGloVe increases the F1 score of 9.8 points on the first file, 12.7 on the second and 25.3 on the third.

[6] http://bit.do/esiaS

	P	R	F1
Stanford NER	82.1	66.1	73.2
Retrained Stanford NER	78.9	79.2	79.1
Neural HistoLevy	85.3	83.3	84.3
Neural Levy	83.7	86.8	85.3
Neural HistoFast	83.9	87.4	85.6
Neural GloVe	83.7	87.9	86.0
Neural FastText	86.3	86.3	86.3
Neural HistoGlove	**86.4**	**88.5**	**87.4**

Table 1: Results of the experiments.

	Stanford NER	**Neural HistoGloVe**
	F1	**F1**
Little Pilgrimage	80.9	90.7
Naples Riviera	73.3	86.0
Rome	55.6	80.9

Table 2: Comparison of F1 in the three test files.

6 Conclusions and Future Works

In this paper we presented the application of a neural architecture to the automatic identification of place names in historical texts. We chose to work on an under-investigated text genre, namely travel writings, that presents a set of specific linguistic features making the NER task particularly challenging. The deep learning approach, combined with in-domain training set and in-domain historical embeddings, outperforms the linear CRF classifier of the Stanford NER module without the need of performing feature engineering. Annotated corpus, best model and historical word vectors are all freely available online.

As for future work, we plan to experiment with a finer-grained classification so to distinguish different types of locations. In addition, another aspect worth studying is the georeferencing of identified place names so to map the geographical dimension of travel writings in Italy. An example of visualisation is given in Figure 1 where the locations automatically identified from the test file taken from the book "Naples Riviera" are displayed: place names have been georeferenced us-

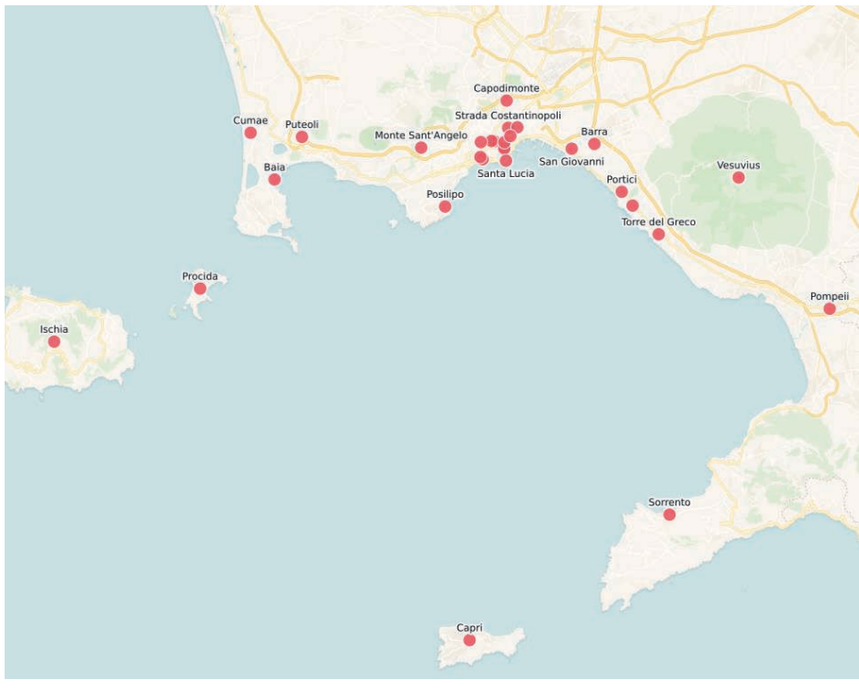


Figure 1: Map of place names in the Neapolitan area mentioned in the "Naples Riviera" test file.

ing the Geocoding API[7] offered by Google and displayed through the Carto[8] web mapping tool. Another interesting work would be the detection of itineraries of past travellers: this application could have a potential impact on the tourism sector, suggesting historical routes alternative to those more beaten and congested and making tourists rediscovering sites long forgotten.

Acknowledgments

The author wants to thank Manuela Speranza for her help with inter-annotator agreement.

References

Tita Beaven. 2007. A life in the sun: Accounts of new lives abroad as intercultural narratives. *Language and Intercultural Communication*, 7(3):188–202.

David J Bodenhamer. 2012. The spatial humanities: space, time and place in the new digital age. In *History in the Digital Age*, pages 35–50. Routledge.

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.

Lars Borin, Dimitrios Kokkinakis, and Leif-Jöran Olsson. 2007. Naming the past: Named entity and animacy recognition in 19th century Swedish literature. In *Proceedings of the Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2007)*, pages 1–8.

Peter Burke. 1997. *Varieties of cultural history*. Cornell University Press.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Mark Davies. 2012. Expanding horizons in historical linguistics with the 400-million word Corpus of Historical American English. *Corpora*, 7(2):121–157.

George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie Strassel, and Ralph M Weischedel. 2004. The Automatic Content Extraction (ACE) Program-Tasks, Data, and Evaluation. In *LREC*, volume 2, pages 837–840.

Maud Ehrmann, Giovanni Colavizza, Yannick Rochat, and Frédéric Kaplan. 2016. Diachronic evaluation of NER systems on old newspapers. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, number EPFL-CONF-221391, pages 97–107. Bochumer Linguistische Arbeitsberichte.

Safaa Eltyeb and Naomie Salim. 2014. Chemical named entities recognition: a review on approaches and applications. *Journal of cheminformatics*, 6(1):17.

[7] https://developers.google.com/maps/documentation/geocoding/start

[8] https://carto.com/

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 on association for computational linguistics*, pages 363–370. Association for Computational Linguistics.

Ian Gregory, Christopher Donaldson, Patricia Murrieta-Flores, and Paul Rayson. 2015. Geoparsing, GIS, and textual analysis: Current developments in spatial humanities research. *International Journal of Humanities and Arts Computing*, 9(1):1–14.

Ralph Grishman and Beth Sundheim. 1996. Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, volume 1.

Maryam Habibi, Leon Weber, Mariana Neves, David Luis Wiegandt, and Ulf Leser. 2017. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, 33(14):i37–i48.

Alison Jones and Gregory Crane. 2006. The challenge of Virginia Banks: an evaluation of named entity analysis in a 19th-century newspaper collection. In *Digital Libraries, 2006. JCDL'06. Proceedings of the 6th ACM/IEEE-CS Joint Conference on*, pages 31–40. IEEE.

Frédéric Kaplan. 2015. A map for big data research in digital humanities. *Frontiers in Digital Humanities*, 2:1.

Omer Levy and Yoav Goldberg. 2014. Dependency-Based Word Embeddings. In *ACL (2)*, pages 302–308.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1064–1074.

Sunghwan Mac Kim and Steve Cassidy. 2015. Finding names in trove: named entity recognition for Australian historical newspapers. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 57–65.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Clemens Neudecker, Lotte Wilms, Wille Jaan Faber, and Theo van Veen. 2014. Large-scale refinement of digital historic newspapers with named entity recognition. In *Proc IFLA Newspapers/GENLOC Pre-Conference Satellite Meeting*.

Clemens Neudecker. 2016. An Open Corpus for Named Entity Recognition in Historic Newspapers. In *LREC*.

Lucia C Passaro, Alessandro Lenci, and Anna Gabbolini. 2017. INFORMed PA: A NER for the Italian Public Administration Domain. In *Fourth Italian Conference on Computational Linguistics CLiC-it 2017*, pages 246–251. Accademia University Press.

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.

Nils Reimers and Iryna Gurevych. 2017a. Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. *arXiv preprint arXiv:1707.06799*.

Nils Reimers and Iryna Gurevych. 2017b. Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348.

Yannick Rochat, Maud Ehrmann, Vincent Buntinx, Cyril Bornet, and Frédéric Kaplan. 2016. Navigating through 200 years of historical newspapers. In *iPRES 2016*, number EPFL-CONF-218707.

Rachele Sprugnoli, Giovanni Moretti, Sara Tonelli, and Stefano Menini. 2016. Fifty years of European history through the lens of computational linguistics: the De Gasperi Project. *Italian Journal of Computational Linguistics*, pages 89–100.

Rachele Sprugnoli, Sara Tonelli, Giovanni Moretti, and Stefano Menini. 2017. A little bit of bella pianura: Detecting Code-Mixing in Historical English Travel Writing. In *Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017)*.

Rachele Sprugnoli. 2018. "Two days we have passed with the ancients...": a Digital Resource of Historical Travel Writings on Italy. In *Book of Abstract of AIUCD 2018 Conference*. AIUCD.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Seth Van Hooland, Max De Wilde, Ruben Verborgh, Thomas Steiner, and Rik Van de Walle. 2013. Exploring entity recognition and disambiguation for cultural heritage collections. *Digital Scholarship in the Humanities*, 30(2):262–279.

Multi-source Transformer for Automatic Post-Editing

Amirhossein Tebbifakhr[1,2], **Ruchit Agrawal**[1,2], **Matteo Negri**[1], **Marco Turchi**[1]
[1] Fondazione Bruno Kessler, Via Sommarive 18, Povo, Trento - Italy
[2] University of Trento, Italy
{atebbifakhr,ragrawal,negri,turchi}@fbk.eu

Abstract

English. Recent approaches to the Automatic Post-editing (APE) of Machine Translation (MT) have shown that best results are obtained by neural multi-source models that correct the raw MT output by also considering information from the corresponding source sentence. In this paper, we pursue this objective by exploiting, for the first time in APE, the Transformer architecture. Our approach is much simpler than the best current solutions, which are based on ensembling multiple models and adding a final hypothesis re-ranking step. We evaluate our Transformer-based system on the English-German data released for the WMT 2017 APE shared task, achieving results that outperform the state of the art with a simpler architecture suitable for industrial applications.

Italiano. *Gli approcci più efficaci alla correzione automatica di errori nella traduzione automatica (Automatic Post-editing – APE) attualmente si basano su modelli neurali multi-source, capaci cioè di sfruttare informazione proveniente sia dalla frase da correggere che dalla frase nella lingua sorgente. Seguendo tale approccio, in questo articolo applichiamo per la prima volta l'architettura Transformer, ottenendo un sistema notevolmente meno complesso rispetto a quelli proposti fino ad ora (i migliori dei quali, basati sulla combinazione di più modelli). Attraverso esperimenti su dati Inglese-Tedesco rilasciati per l'APE task a WMT 2017, dimostriamo che, oltre a tale guadagno in termini di semplicità, il metodo proposto ottiene risultati superiori allo stato dell'arte.*

1 Introduction

Automatic post-editing (APE) (Simard et al., 2007b; Simard et al., 2007a; Simard et al., 2009) is the task of fixing errors in a machine-translated text by learning from human corrections. It has shown to be useful for various tasks like domain adaptation (Isabelle et al., 2007) and for reducing time, effort and the overall costs of human translation in industry environments (Aziz et al., 2012).

Recent approaches to the task have shown that better results can be obtained by neural multi-source models that perform the automatic correction of raw MT output by also considering information from the corresponding source sentence (Chatterjee et al., 2015; Pal et al., 2016). However, state-of-the-art APE solutions employ pipelined architectures (Bojar et al., 2017) whose complexity reduces their usability in industrial settings. Indeed, current top systems typically rely on ensembling multiple recurrent neural networks (RNNs) and performing a final re-ranking step (Chatterjee et al., 2017) to select the most promising correction hypothesis. Though competitive, such architectures require training and maintaining multiple components, involving costs that reduce their appeal from the industry perspective.

In this paper, we address this issue, aiming at a method that is suitable for industry applications, in which a single trainable network is preferable to multiple, independently-trained components. Our main contributions are the following:

- We introduce, for the first time in APE, a Transformer-based architecture (Vaswani et al., 2017) that considerably reduces system complexity (thus being efficient and easy to train and maintain);
- In doing so, we modify the Transformer architecture to incorporate multiple encoders, thereby considering also source-side information to increase correction accuracy;

- On shared data sets, we report evaluation results that are comparable (less than 0.5 BLEU score points in the worst case) to those of computationally-intensive state-of-the-art systems based on model ensembling and hypothesis reranking.

2 Methodology

In this Section we shortly overview our approach, by first motivating the use of Transformer (Vaswani et al., 2017) and then by introducing our modifications to deploy it for APE.

Most of the competitive neural approaches in machine translation employ deep recurrent networks (Sutskever et al., 2014; Bahdanau et al., 2015). These approaches follow the encoder-decoder architecture. A sequence of words [x_1, x_2, .. , x_n] is given to an encoder, which maps it to a sequence of continuous representations, i.e. the hidden state of the encoder. At each time step, based on these continuous representations and the generated word in the previous time step, a decoder generates the next word. This process continues until the decoder generates the end-of-the-sentence word. More formally, the decoder predicts the next word y_t, given the context vector c and the previously predicted words y_1 to y_{t-1} by defining a probability over the translation **y** as follows:

$$p(\mathbf{y}) = \prod_{t=1}^{T} p(y_t|[y_1, .., y_{t-1}], c) \quad (1)$$

The context vector c is a weighted sum computed over the hidden states of the encoder. The weights used to compute the context vector are obtained by a network called attention model that finds an alignment between the target and source words (Bahdanau et al., 2015). From an efficiency standpoint, a major drawback of these approaches is that, at each time step, the decoder needs the hidden state of the previous time step, thus hindering parallelization. Other approaches have been proposed to avoid this sequential dependency (e.g. using convolution as a main building blocks) and make parallelization possible (Gehring et al., 2017; Kalchbrenner et al., 2016). Although they can avoid the recurrence, they are not able to properly learn the long term dependencies between words.

The Transformer architecture, introduced in (Vaswani et al., 2017), set a new state-of-the-art in NMT by completely avoiding both recurrence and convolution. Since the model does not leverage the order of words, it adds positional encoding to the word embeddings to enable the model to capture the order. In Transformer, the attention employed is a multi-headed self-attention, which is a mapping from (query, key, value) tuples to an output vector. The self-attention is defined as follows:

$$SA(Q, K, V) = softmax(QK^T/\sqrt{d_k})V \quad (2)$$

where Q is the query matrix, K is the key matrix and V is the value matrix, d_k is the dimensionality of the queries and keys, and SA is the computed self-attention.

The multi-head attention is computed as follows:

$$MH(Q, K, V) = Concat(head_1, ..., head_h)W^O \quad (3)$$

where MH is the multi-head attention, h is the number of attention layers (also called "heads"), $head_i$ is the self-attention computed over the i^{th} attention layer and W^O is the parameter matrix of dimension $hd_v{*}d_{model}$. The encoder layers consist of a multi-head self-attention, followed by a position-wise feed forward network. In the self-attention, the queries, keys and values matrices come from the previous layer. In the decoder, the layers have an extra encoder-decoder multi-head attention after the multi-head self-attention, where the key and value matrices come from the encoder and the query matrix comes from the previous layer in the decoder. Also, inputs to the multi-head self-attention in the decoder are masked in order to not attend to the next positions. Finally, a softmax normalization is applied to the output of the last layer in the decoder to generate a probability distribution over the target vocabulary.

In order to encode the source sentence in addition to the MT output, we employ the multi-source method (Zoph and Knight, 2016), wherein the model is comprised of separated encoders (with a different set parameters) to capture the source sentence and the MT output respectively. For the Transformer, we concatenate the two encoder outputs and that is passed as the key in the attention. This helps for a better representation, in turn leading to more effective attention during decoding time.

train			development	test	
synthetic 4M	synthetic 500K	in-domain	in-domain	in-domain 2016	in-domain 2017
4,391,180	526,368	23,000	1,000	2,000	2,000

Table 1: Statistics for synthetic and in-domain datasets

3 Experiment Setup

3.1 Data

For the sake of a fair comparison with the best performing system at the WMT 2017 APE shared task (Chatterjee et al., 2017), we use the same training, development and test WMT datasets. The training data consists of three different corpora. One of them is released by the task organizers and contains 23K triplets from the Information Technology domain. The other two are synthetic data created by (Junczys-Dowmunt and Grundkiewicz, 2017). They respectively contain ∼4M and ∼500K English-German triplets generated by a round-trip translation process. By using two phrase-based translation models, German-English and English-German, German monolingual data are first translated into English and then the obtained outputs are translated back into German. The original German monolingual data are considered as post-edits, the English translated data are considered as source sentences, and the German back-translated data are considered as machine translation outputs. The development set is the one released for WMT 2017 APE shared task, which contains 1K in-domain triplets. We evaluate our model using the two test sets released for WMT 2016 and 2017 APE shared tasks, each containing 2K in-domain triplets. Table 1 summarizes the statistics of the datasets. To avoid unknown words and to keep under control the vocabulary size, we apply byte pair encoding (Sennrich et al., 2016) to all the data.

3.2 Evaluation Metrics

For evaluation, we use the two official metrics of the WMT APE task: i) TER (Snover et al., 2006) which is based on edit distance and ii) BLEU, which is the geometric mean of n-gram precision (Papineni et al., 2002). They are both applied on tokenized and true-cased data.

3.3 Term of Comparison

We compare the performance of our Transformer model with two baselines: i) **MT Baseline**: the output of a "*do-nothing*" APE model that leaves all the original MT outputs untouched, and ii) **Ens8 + RR**: the winning system at the WMT 2017 APE shared task (Chatterjee et al., 2017). It comprises 4 different models based on RNN architecture:

- **SRC_PE** a single-source model that exploits only the source sentence to generate post-edits;
- **MT_PE** a single-source model that only exploits the machine translation output to generate post-edits;
- **MT+SRC_PE** a multi-source model that exploits both the source sentence and the MT output to generate post-edits;
- **MT+SRC_PE_TSL** another multi-source model with a task-specific loss function in order to avoid over correction.

For mixing the context vectors of the two encoders, Ens8 + RR uses a merging layer. This layer applies a linear transformation over the concatenation of the two context vectors. Chatterjee et al. (2017) compared the performance of these 4 models on the development set, and reported that MT+SRC_PE outperforms the other models. They also ensembled the two best models for each configuration to leverage all the models in a single decoder. On top of that, they also trained a re-ranker (Pal et al., 2017) to re-order the n-best hypotheses generated by this ensemble. In order to train the re-ranker, they used a set of features which are mainly based on edit distance. This set includes number of insertions, deletions, substitutions, shifts, and length ratios between MT output and APE hypotheses. It also includes precision and recall of the APE hypotheses. In Section 4, we compare our model with the SRC+MT_PE model and the ensembled model plus re-ranker (Ens8+RR). We train these models with the same settings reported in (Chatterjee et al., 2017).

3.4 System Setting

We initially train a generic Transformer model by using the ∼4M synthetic data. Then, we fine-tune

Systems	TER	BLEU
Baseline	24.81	62.92
SRC+MT_PE	19.77	70.72
Ens8 + RR	19.22	71.89
Transformer	19.17	71.58
Avg4	18.77	72.04

Table 2: performance of APE systems on 2017 development dataset (*en-de*)

Systems	Test2016		Test2017	
	TER	**BLEU**	**TER**	**BLEU**
MT Baseline	24.76	62.11	24.48	62.49
Ens8 + RR	19.32	70.88	19.60	70.07
Transformer	19.25	70.70	19.81	69.64
Avg4	18.79	71.48	19.54	70.09

Table 3: performance of APE systems on 2016 and 2017 test datasets (*en-de*)

the resulting model on the union of the ∼500K and the in-domain training data (multiplied 20). Our Transformer model uses word embedding with 512 dimensions. The decoder and each encoder have 4 attention layers with 512 units, 4 parallel attention heads, and a feed-forward layer with 1,024 dimensions. The network parameters are updated using Lazy Adam optimizer (Kingma and Ba, 2014), with mini-batch size of 8,192 tokens for generic training and 2,048 tokens for fine-tuning. The learning rate is varied using a warm-up strategy (Vaswani et al., 2017) with warm-up steps equal to 8,000. During training, the dropout rate and the label smoothing value are set to 0.1. During decoding, we employ beam search with beam width equal to 10. For both the generic and fine-tuning steps, we continue the training for 10 epochs and choose the best model checkpoints based on their performance on the development set. For our implementation, we use the OpenNMT-tf toolkit (Klein et al., 2017).

4 Results and Discussion

Table 2 shows the results obtained by different models on the development set. Together with our simple Transformer model (Transformer), it also reports the performance of averaging the weights of the 4 best model checkpoints (Avg4). Our Transformer model performs better than the SRC+MT_PE model (-0.6 TER and +0.86 BLEU) showing that using the Transformer architecture instead of RNN is helpful. Also, our Transformer model outperforms Ens8+RR in terms of TER, with only a small loss in terms of BLEU. This highlights that our simple model can achieve comparable results with the best performing systems, but using less complex architecture. By averaging different Transformer checkpoints, our model outperforms Ens8+RR by -0.45 TER and +0.15 BLEU. This gain confirms the results reported by Popel and Bojar (2018), who showed that averaging the model's checkpoints weights is advantageous. Moreover, we are not loosing our simplicity in comparison with ensembling, since we are choosing the model's checkpoints in a single training round and this does not require training several models and architectures. In order to confirm our observation on the development set, we also evaluated our model in compare to Ens8+RR on the two test sets. Table 3 shows the results obtained on the two test sets, which confirm our observations on development data. The averaged model has the best performance over the RNN systems and single Transformer. It significantly outperforms Ens8+RR on 2016 test data, while a marginal improvements is obtained on the 2017 test set. To conclude, our results confirm the trend seen in Machine Translation, where Transformer outperforms RNN-based systems on different language pairs and datasets using a simpler architecture. Beside this, our extension targeting the inclusion of source-side information sets a new state of the art in APE.

5 Conclusion

We developed and used a multi-source Transformer architecture for neural Automatic Post-editing. In contrast to the current state-of-the-art systems for APE, which are based on RNN architectures that typically comprise multiple components, we used a single model which can be trained in an end-to-end fashion. This solution is particularly suitable for industrial sectors, where maintaining different components is costly and inefficient. Our experiments show that our simplest model has comparable results to the best RNN systems, while the best one can even perform slightly better. This sets the new state of the art in APE and confirms the superiority of Transformer in sequence-to-sequence learning tasks.

References

Wilker Aziz, Sheila Castilho, and Lucia Specia. 2012. Pet: a tool for post-editing and assessing machine translation. In *LREC*, pages 3982–3987.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the Second Conference on Machine Translation*, pages 169–214. Association for Computational Linguistics.

Rajen Chatterjee, Marion Weller, Matteo Negri, and Marco Turchi. 2015. Exploring the planet of the apes: a comparative study of state-of-the-art methods for mt automatic post-editing. 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)*, volume 2, pages 156–161.

Rajen Chatterjee, M. Amin Farajian, Matteo Negri, Marco Turchi, Ankit Srivastava, and Santanu Pal. 2017. Multi-source neural automatic post-editing: Fbk's participation in the wmt 2017 ape shared task. In *Proceedings of the Second Conference on Machine Translation*, pages 630–638. Association for Computational Linguistics.

Jonas Gehring, Michael Auli, David Grangier, and Yann Dauphin. 2017. A convolutional encoder model for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 123–135. Association for Computational Linguistics.

Pierre Isabelle, Cyril Goutte, and Michel Simard. 2007. Domain adaptation of mt systems through automatic post-editing.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2017. The amu-uedin submission to the wmt 2017 shared task on automatic post-editing. In *Proceedings of the Second Conference on Machine Translation*, pages 639–646. Association for Computational Linguistics.

Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. 2016. Neural machine translation in linear time. *arXiv preprint arXiv:1610.10099*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72. Association for Computational Linguistics.

Santanu Pal, Sudip Kumar Naskar, Mihaela Vela, and Josef van Genabith. 2016. A neural network based approach to automatic post-editing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 281–286.

Santanu Pal, Sudip Kumar Naskar, Mihaela Vela, Qun Liu, and Josef van Genabith. 2017. Neural automatic post-editing using prior alignment and reranking. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 349–355. Association for Computational Linguistics.

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 on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Martin Popel and Ondřej Bojar. 2018. Training tips for the transformer model. *The Prague Bulletin of Mathematical Linguistics*, 110(1):43–70.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

Michel Simard, Cyril Goutte, and Pierre Isabelle. 2007a. Statistical phrase-based post-editing.

Michel Simard, Nicola Ueffing, Pierre Isabelle, and Roland Kuhn. 2007b. Rule-based translation with statistical phrase-based post-editing. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 203–206. Association for Computational Linguistics.

Michel Simard, Pierre Isabelle, George Foster, Cyril Goutte, and Roland Kuhn. 2009. Means and method for automatic post-editing of translations, December 31. US Patent App. 12/448,859.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts, USA.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz 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 30*, pages 5998–6008. Curran Associates, Inc.

Barret Zoph and Kevin Knight. 2016. Multi-source neural translation. *arXiv preprint arXiv:1601.00710*.

Classifying Italian newspaper text: news or editorial?

Pietro Totis
Università degli Studi di Udine
totis.pietro@spes.uniud.it

Manfred Stede
Applied Computational Linguistics
University of Potsdam, Germany
stede@uni-potsdam.de

Abstract

English. We present a text classifier that can distinguish Italian news stories from editorials. Inspired by earlier work on English, we built a suitable train/test corpus and implemented a range of features, which can predict the distinction with an accuracy of 89,12%. As demonstrated by the earlier work, such a feature-based approach outperforms simple bag-of-words models when being transferred to new domains. We argue that the technique can also be used to distinguish opinionated from non-opinionated text outside of the realm of newspapers.

Italiano. *Presentiamo una tecnica per la classificazione di articoli di giornale in italiano come articoli di cronaca oppure editoriali. Ispirandoci a precedenti pubblicazioni riguardanti la lingua inglese, abbiamo costruito un corpus adatto allo scopo e selezionato un insieme di caratteristiche testuali in grado di distinguere il genere con un accuratezza dell' 89,12%. Come dimostrato dai lavori precedenti, questo approccio basato sulle proprietà del testo mostra risultati migliori rispetto ad altri quando trasferito a nuovi argomenti. Riteniamo inoltre che questa tecnica possa essere usata con successo anche in contesti diversi dagli articoli di giornale per distinguere testi contenenti opinioni dell'autore e non.*

1 Introduction

The computational task of text classification is typically targeting the question of *domain*: Is a text about sports, the economy, local politics, etc. But texts can also be grouped by their *genre*: Is it a business letter, a personal homepage, a cooking recipe, and so on. In this paper, we perform genre classification on newspaper text and are specifically interested in the question whether a text communicates a news report or gives an opinion, i.e., it is an editorial (or some similar opinionated piece). This task is relevant for many information extraction applications based on newspaper text, and it can also be extended from newspapers to other kinds of text, where the distinction "opinionated or not" is of interest, as in sentiment analysis or argumentation mining.

Our starting point is the work by (Krüger et al., 2017), who presented a news/editorial classifier for English. They demonstrated that using linguistically-motivated features leads to better results than bag-of-words or POS-based models, when it comes to changing the domain of text (which newspaper, which time of origin, which type of content). To transfer the approach to Italian, we assembled a suitable corpus for training and testing, selected preprocessing tools, and adapted the features used by the classifier from Krüger et al. Our results are in same range of the original work, indicating that the problem can be solved for Italian in pretty much the same way. We found some differences in the relative feature strengths, however.

After considering related work in Section 2, we describe our corpus (Section 3) and the classification experiments (Section 4), and then conclude.

2 Related Work

In early work, (Karlgren and Cutting, 1994) ran genre classification experiments on the Brown Corpus and employed the distribution of POS-tags as well as surface-based features such as length of words, sentences and documents, type/token ratio, and the frequency of the words 'therefore', 'I', 'me', 'it', 'that' and 'which'. Among the experiments, the classification of 'press editorial'

yielded 30% errors, and that of 'press reportage' 25%. On the same data, (Kessler et al., 1997) used additional lexical features (latinate affixes, date expressions, etc.) and punctuation. The authors reported these accuracies: reportage 83%, editorial 61%, scitech 83%, legal 20%, nonfiction (= other expository writing) 47%, fiction 94%.

The alternative method is to refrain from any linguistic analysis and instead use bag-of-tokens (2003), bag-of-words (Freund et al., 2006), (Finn and Kushmerick, 2003) or bag-of-character-n-gram (Sharoff et al., 2010) models. This has the obvious advantage of knowledge-freeness and yields very good results in the domains of the training data, but, as found for instance by Finn and Kushmerick, a bag-of-words model performs very badly in cross-domain experiments. Likewise, (Petrenz and Webber, 2011) show in their replication experiments that this idea is highly vulnerable to topic/domain shifting: the models largely learn from the content words in the training texts, and these can be very different from day to day, when the news and the opinions on them reflect the current affairs.

(Toprak and Gurevych, 2009) experimented with various lexical features: Word-based features included unigrams, bigrams, variants with surrounding tokens, as well as frequency-amended lemma features (using a tf*idf measure); lexicon features exploited the Subjectivity Clues Lexicon (Wilson et al., 2005), SentiWordnet (Esuli and Sebastiani, 2006), and a list of communication and mental verbs. It turned out that word class features outperform the other classes, with an accuracy of up to 0.857. Specifically, the tf*idf representation was successful. Such frequency-based representations are known to be effective for classical topic categorization tasks, and this study provides an indication that they may also help for related tasks (especially when the class distribution is skewed). Another finding was that plain unigrams beat the larger n-grams and certain context features.

(Cimino et al., 2017) investigated the role of different feature types in the task of Automatic Genre Classification. In this study a set of relevant features is extracted across different linguistic description levels (lexical, morpho-syntactic and syntactic) and a meaningful subset is then selected through an incremental feature selection procedure. The results show that syntactic features are the most effective in order to discriminate between different text genres.

Finally, as mentioned earlier, we build our work on that of (Krüger et al., 2017), who systematically tested a meaningful set of linguistic features. Among several classifiers from the `WEKA` libraries, the SMO classifiers performed best, and the models based on linguistic features outperformed standard bag-of-lemma approaches across different genres, but the latter still performed very well on the same genre on which they were trained. Krüger et al. then tested which features are most predictive for each class, and related these observations to their original expectations.

3 Dataset

For our study, we built a corpus of about 1000 Italian newspaper articles, which are equally divided into editorials and news articles.

The editorials have been collected from the website of the Italian newspaper *"Il Manifesto"* and we removed headers and footers that serve as metadata for the newspaper, such as "2017 IL NUOVO MANIFESTO SOCIETÀ COOP. EDITRICE". The news articles are from the Adige corpus[1], a collection of news stories from the local newspaper *L'Adige* categorized into different topics of news, such as *sport*, *finance* or *culture*. The corpus is also annotated with semantic information related to temporal expressions and entities. However, we have not exploited these features since they were not available on the editorials.

Both corpora have been annotated using the *TreeTagger* tool[2] (Schmid, 1994), which provides an annotation of the form WORD, POS-TAG, LEMMA.

In order to reproduce the types of classification features used by (Krüger et al., 2017), some lexical resources are needed. The corresponding Italian vocabulary has been collected from different sources:

- A list of connectives, categorized into temporal, causal, contrastive and expansive connectives, has been obtained from LICO (Feltracco et al., 2016), a lexicon for Italian connectives.

[1] `http://ontotext.fbk.eu/icab.html`

[2] Future improvements include using a more modern postagger such as UDPipe: `https://ufal.mff.cuni.cz/udpipe`

	Acc.	Prec.	Recall	F1
L	83,35	86,04	79,42	82,60
P	84,49	85,80	82,50	84,11
U	82,29	80,29	85,38	82,75
L+U	87,75	88,88	86,15	87,50
L+P	87,27	88,46	85,58	87,00
U+P	87,37	87,31	87,31	87,31
L+P+U	89,09	89,64	88,27	88,95

Table 1: Linear SMO results: L: Linguistic features, P: POS tagging, U: Unigrams

- A list of communication verbs (*say, argue, state*, etc.) has been obtained from the lexical database *MultiWordNet*[3] for a total of 54 entries.
- Sentiment features rely on the *Sentix*[4] lexicon for Italian sentiment analysis, which assigns to each lemma a positive and negative score, plus a score of polarity and intensity.

4 Experiments

Feature	Weight
LING:PRONOUNS	3,5452
LING:TEMPORALCONN	2,0647
LING:SENT_POS	1,8040
LING:NEGATIONS	1,7301
LING:SENT_NEG	1,6609
LING:PAST	1,3686
LING:CONTRASTIVECONN	1,2816
LING:INFINITIVE	1,2230
LING:SENT_ADJ_POL	1,2114
LING:SENT_ADJ_NEG	1,0880
LING:CONDIMP	1,0796
LING:GERUND	1,0653
LING:COMMAS	0,9658
LING:SENT_INT	0,9593
LING:IMPERFECT	0,7801

Table 2: Linguistic features pointing to opinionated text

4.1 Main experiment: feature performance

In our experiments, we were primarily interested in comparing the accuracies obtained by (i) linguistic features, (ii), unigram counts, (iii) part of speech tags counts, and their combinations as indicators for classifying the newspaper articles from the dataset. Four different classifiers from the WEKA library have been tested: linear and polynomial SMO (kernel with $e = 2$), J48 trees and Naive Bayes classifier, with a 10-fold cross-validation evaluation. The SMO classifiers proved to be the most accurate, with the polynomial SMO having marginally higher scores than the linear counterpart. In Table 1 we provide our results obtained with that approach. It can be seen that combining feature sets generally outperforms the individual sets, and in fact the combination of all three yields the best results.

Our set of linguistic features was modeled closely after that of Krüger et al., because we wanted to know how well it can be transferred to languages other than English. These features can be summarized as follows: text statistics (length of a sentence, frequency of digits, etc.); ratio of punctuation symbols; ratio of temporal, causal and other connectives; verb tenses; pronouns (esp. 1^{st} and 2^{nd} person) and sentiment indicators.

	Acc.	Prec.	Recall	F1
L	83,90	84,21	82,75	83,47
P	64,71	63,08	69,49	66,12
U	39,17	43,30	70,00	53,50
L+U	65,00	50,57	73,33	59,86
L+P	72,57	70,37	71,70	71,03
U+P	50,83	50,57	73,33	59,86
L+P+U	61,34	57,83	81,35	67,60

Table 3: Linear SMO results on Amazon reviews and Wikipedia articles

Feature	Weight
LING:CITATIONS	4,8912
LING:COMPLEXITY	2,6676
LING:PASTPERFECT	2,1070
LING:FUTURE	2,0092
LING:TOKENLENGTH	1,8754
LING:CAUSALCONN	1,7568
LING:SENT_POL	0,9710
LING:VOS	0,7414
LING:IMPERATIVE	0,6871
LING:FSPRONOUNS	0,6518
LING:FPRONOUNS	0,6518
LING:MODALS	0,4237

Table 4: Linguistic features pointing to news text

[3] http://multiwordnet.fbk.eu/english/home.php

[4] http://valeriobasile.github.io/twita/sentix.html

The set also includes the presence of modal verbs and negation operators, morphological features of the matrix verb (tense, mood), as well as some selected part-of speech and basic text statistic features, as they had already been proposed in the early related work.

The feature weights assigned by the linear classifier are shown in tables 2 and 4 in order to highlight which linguistic features represent good indicators towards one or another type of article, and with how much strength.

The results obtained offer interesting analogies with the English corpus analysed by (Krüger et al., 2017). For instance, pronouns, negations and sentiment represent strong indicators for opinionated texts, while complexity, future, communication verbs, token length and causal connectives are all features pointing towards news reports in both languages. An interesting difference is the role of past tense, which for English had been found to correlate more with news than with editorials, and here it plays a different role.

4.2 Testing domain change robustness

We then evaluated another aspect of the task, viz. domain robustness: we split the news corpus into a training set (categories Attualità, Sport and Economia) and a test set (categories Cultura and Trento) in order to evaluate the robustness of the classifier when unseen categories are submitted. All the classification performances in this setting show a drop of performance of only about 0,03%, demonstrating that the classification performances are not overfitted to the topics of the articles.

Finally, to further test domain change robustness, we tested the classifier – with the model trained on the newspaper corpora – on a set of 60 Amazon reviews versus 60 Wikipedia articles (all randomly chosen). As the results in Table 3 show, the linguistic features perform remarkably robust also on this quite different data. The bad results for unigrams on the one hand are not so surprising, but they have to be taken with a grain of salt, because we employed the same low frequency filtering as in the main experiment: unigrams that occur less than five times are not being considered, in order to reduce the feature space. This might well lead to poorer results for a small data set like the 120 texts used here.

4.3 Replication

Altough we cannot make public all the data we used in this experiment, we uploaded our code on a public repository[5] to provide a description of our implementation.

5 Conclusion

We presented, to our knowledge, the first classifier that is able to distinguish 'news' from 'editorials' in an Italian newspaper corpus. It follows a linguistic feature-oriented approach proposed by (Krüger et al., 2017) for English, who had demonstrated that it outperforms lexical and POS-based models. In our implementation, With an accuracy of 89.09% the distinction between the two subgenres can be drawn quite reliably. Our results are comparable to that of Krüger et al., which indicates (again, to our knowledge for the first time) that their feature space is applicable successfully to languages other than English.

Our central concern for this kind of task is robustness against domain changes of different kinds. To this end, Krüger et al. had worked with different newspaper sources and demonstrated the utility of the feature approach in such settings. While we were not able to assemble large corpora from different papers, we ran other experiments in the same vein, where the first shows that the system is robust against changing the portions of the newspapers (i.e., economy versus local affairs, and so on). In the second one, we applied the classifier, as trained on the newspaper data, to the distinction between Italian Wikipedia articles and Amazon reviews, where the results remained stable as well. We take this as an indication that the classifier captures a general difference between 'opinionated' and 'non-opinionated' text, and not just some 'ad hoc' phenomena of certain newspaper sub-genres.

References

[Cimino et al.2017] Andrea Cimino, Martijn Wieling, Felice Dell'Orletta, Simonetta Montemagni, and Giulia Venturi. 2017. Identifying predictive features for textual genre classification: the key role of syntax. 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-*

[5] https://bitbucket.org/PietroTotis/classifying-italian-newspaper-text-news-or-editorial/src/master/

13, 2017., volume 2006 of *CEUR Workshop Proceedings*. CEUR-WS.org.

[Esuli and Sebastiani2006] Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06*, pages 417–422.

[Feltracco et al.2016] Anna Feltracco, Elisabetta Jezek, Bernardo Magnini, and Manfred Stede. 2016. Lico: a lexicon of italian connectives. In *Proceedings of the 3rd Italian Conference on Computational Linguistics (CLiC-it)*, Napoli.

[Finn and Kushmerick2003] Aidan Finn and Nicholas Kushmerick. 2003. Learning to classify documents according to genre. *IJCAI-03 Workshop on Computational Approaches to Style Analysis and Synthesis*.

[Freund et al.2006] Luanne Freund, Charles L. A. Clarke, and Elaine G. Toms. 2006. Towards genre classification for IR in the workplace. In *Proceedings of the 1st international conference on Information interaction in context*, IIiX, pages 30–36, New York, NY, USA. ACM.

[Karlgren and Cutting1994] Jussi Karlgren and Douglass Cutting. 1994. Recognizing text genres with simple metrics using discriminant analysis. In *Proceedings of the 15th conference on Computational linguistics - Volume 2*, COLING '94, pages 1071–1075, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Kessler et al.1997] Brett Kessler, Geoffrey Numberg, and Hinrich Schütze. 1997. Automatic detection of text genre. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 32–38. Association for Computational Linguistics.

[Krüger et al.2017] Katarina R. Krüger, Anna Lukowiak, Jonathan Sonntag, Saskia Warzecha, and Manfred Stede. 2017. Classifying news versus opinions in newspapers: Linguistic features for domain independence. *Natural Language Engineering*, 23(5):687–707.

[Petrenz and Webber2011] Philipp Petrenz and Bonnie Webber. 2011. Stable classification of text genres. *Computational Linguistics*, 37(2):385–393.

[Schmid1994] Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, Manchester.

[Sharoff et al.2010] Serge Sharoff, Zhili Wu, and Katja Markert. 2010. The Web Library of Babel: evaluating genre collections. *Proceedings of the Seventh Conference on International Language Resources and Evaluation*, pages 3063–3070.

[Toprak and Gurevych2009] Cigdem Toprak and Iryna Gurevych. 2009. Document level subjectivity classification experiments in deft'09 challenge. In *Proceedings of the DEFT'09 Text Mining Challenge*, pages 89–97, Paris, France.

[Wilson et al.2005] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of HLT-EMNLP-2005*.

[Yu and Hatzivassiloglou2003] Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 129–136. Association for Computational Linguistics.

Multi-Word Expressions in spoken language: PoliSdict

Daniela Trotta[1], **Michele Stingo**[2] **Raffaele Guarasci**[3] **Annibale Elia**[1]
Teresa Albanese[1]

[1] Università di Salerno, Salerno, Italy
[2] Network Contacts, Molfetta (BA), Italy
[3] ICAR, Consiglio Nazionale delle Ricerche, Italy

{dtrotta,talbanese}@unisa.it michele.stingo@network-contacts.it raffaele.guarasci@icar.cnr.it elia@unisa.it

Abstract

English. *The term multiword expressions (MWEs) is referred-to a group of words with a unitary meaning, not inferred from that of the words that compose it, both in current use and in technical-specialized languages. In this paper, we describe PoliSdict an Italian electronic dictionary composed of multi-word expressions (MWEs) automatically extracted from a multimodal corpus grounded on political speech language, currently being developed at the "Maurice Gross" Laboratory of the Department of Political Sciences, Social and Communication of the University of Salerno, thanks to a loan from the company Network Contacts. We introduce the methodology of creation and the first results of a systematic analysis which considered terminological labels, frequency labels, recurring syntactic patterns, further proposing an associated ontology.*

Italiano. *Con il termine polirematica si fa generalmente riferimento ad un gruppo di parole con significato unitario, non desumibile da quello delle parole che lo compongono, sia nell'uso corrente sia in linguaggi tecnico-specialistici. In questo contributo viene presentato PoliSdict un dizionario elettronico in lingua italiana composto da espressioni polirematiche occorrenti nel parlato spontaneo estratte a partire da un corpus multimodale di dominio politico in lingua italiana in corso di ampliamento presso il Laboratorio "Maurice Gross" del Dipartimento di Scienze Politiche, Sociali e della Comunicazione dell'Università degli Studi di Salerno, grazie a un finanziamento della società Network Contacts. Viene presentata la metodologia di creazione ed i primi risultati di un'analisi sistematica che ha considerato etichette terminologiche, marche d'uso e pattern ricorrenti, proponendo infine un'ontologia associata.*

1 Introduction

The term multi-word expressions (MWEs) includes a wide range of constructions such as noun compounds, adverbials, binomials, verb particles constructions, collocations, and idioms (Vietri, 2014). D'Agostino & Elia (1998) consider MWUs part of a continuum in which combinations can vary from a high degree of variability of co-occurrence of words (combinations with free distribution), to the absence of variability of co-occurrence[1]. They identify four different types of combinations of phrases or sentences, namely (i) with a high degree of variability of co-occurrence among words; (ii) with a limited degree of variability of co-occurrence among words; (iii) with no or almost no variability of co-occurrence among words; (iv) with no variability of co-occurrence among words. The essential role played by MWEs in Natural Language Processing (NLP) and linguistic analysis in general has been long recognised, as confirmed by then numerous dedicated workshops and special issues of journals discussing this subject in recent years (CSL, 2005; JLRE, 2009), and this appears more clear if we consider as the detection of MWEs represents a real issue in several NLP tasks such as semantic parsing and machine translation (Fellbaum, 2011). According to Chiari (2012) regarding the Italian language a line of great

[1] Concerning compositionality, the study of Nunberg et al. (1994) is noteworthy. This study undermines the issue of compositionality, as widely emphasized in Vietri (2014).

interest is represented by the works of Annibale Elia and Simonetta Vietri (Elia, D'Agostino et al 1985, Vietri 1986, D'Agostino and Elia 1998, Vietri 2004). Finally the discussion concerning the MWEs in Italian lexicography has been systematized in the GRADIT (De Mauro 1999) which records 132.000 different MWEs, whose collection was coordinated by Annibale Elia at the Department of Communication Sciences of the University of Salerno. This research is part of the larger project BIG 4 M.A.S.S. conducted by the company Network Contacts[2] in collaboration with the Department of Social Politics and Communication, which received funding to develop semantic and syntactic modules of Italian.

2 Related work

In the last twenty years or so MWEs have been an increasingly important concern for NLP. MWEs have been studied for decades in phraseology under the term phraseological unit. But in the early 1990s, MWEs received increasing attention in corpus-based computational linguistics and NLP. Early influential work on MWEs includes Smadja (1993), Dagan and Church (1994), Wu (1997), Daille (1995), Wermter and Chen (1997), McEnery et al. (1997), and Michiels and Dufour (1998). These studies address the automatic treatment of MWEs and their applications in practical NLP and information systems. An important research contribution is the Multiword Expression Project carried out at Stanford University, which began in 2001 to investigate means to encode a variety of MWEs in precision grammars [3] . Other major work has been conducted at Lancaster University, which resulted in a large collection of semantically annotated English, Finnish and Russian MWE dictionary resources for a semantic annotation tool (Rayson et al. 2004; Lo¨fberg et al. 2005; Piao et al. 2005; Mudraya et al. 2006). Since then, many advances have been made, either looking at MWEs in general (Zhang et al., 2006; Villavicencio et al., 2007), or focusing on specific MWE types, such as collocations (Pearce, 2002), phrasal verbs (Baldwin, 2005; Ramisch et al., 2008) or compound nouns (Keller et al., 2002). A popular type-independent alternative to MWE identification is to use statistical AMs (Evert and Krenn, 2005; Zhang et al., 2006; Villavicencio et al., 2007). Concerned MWE identification and extraction from monolingual corpora, Kim and Baldwin (2006) proposed a method for automatically identifying English verb particle constructions (VPCs), Pecina (2009) reported an evaluation of a set of lexical association measures based on the Prague Dependency Treebank and the Czech National Corpus, Strik et al. (2010) investigated the possible ways of automatically identifying Dutch MWEs in speech corpora. Related to lexical representation of MWEs in a lexicon and a syntactic treebank, Gregoire (2010) discusses the design and implementation of a Dutch Electronic Lexicon of Multiword Expressions (DuELME), which contains over 5,000 Dutch multiword expressions. Bejc˘ek and Stranak (2010) describe the annotation of multiword expressions found within the Prague Dependency Treebank. In NLP, MWEs in spoken language have been studied in the field of automatic speech recognition, generally with the aim of establishing to what extent modeling such expressions can help reducing word error rate (Strik and Cucchiarini 1999). So a review of related work about MWEs highlights the lack of electronic dictionaries of Italian MWEs for spoken language, hence the idea of creating an *ad hoc* dictionary starting from a resource of political domain. That being said, it should be specified here that this study represents an initial experiment on a relatively small sample, since a larger balanced corpus would be necessary for a broader coverage. Political discourse offers interesting cues for analysis and experimentation (Frank, 1996; Dixon, 2002; Callander & Wilkie, 2007; Osborne, 2014). In recent years, political speech has earned much attention (Guerini et al., 2008; 2013; Esposito et al., 2015) for purposes, ranging from analysis of communication strategies (Muelle, 1973; Wilson, 1990; Wilson, 2011), persuasive Natural Language Processing, politicians' rhetoric (Stover & Ibroscheva, 2017) and virality of information diffusion (Caliandro & Balina, 2015). Regarding MWs resources for Italian we may mention recent contributions such as PANACEA (Platform for Automatic, Normalized Annotation and Cost-Effective Acquisition of Language Resources for Human Language

[2] Network Contacts, is one of the national leader players in the areas of BPO (business process outsourcing), CRM (customer relationship management), Digital Interaction and Call&Contact Center services. Over the years, it has built numerous partnership with some of the most recognized national academic players, such as the University of Salerno, so as to face stimulating research challenges in the fields of Artificial Intelligence and Natural Language Processing.

[3] For more information cfr. http://mwe.stanford.edu

Techologies) that includes Italian word n-grams and Italian word/tag/lemma n-grams in the "Labour" (LAB) domain (Bel at al., 2012) and also PARSEME-IT Corpus, an annotated Corpus of Verbal Multiword Expressions in Italian (Monti et al., 2017).

3 PoliSdict

According to Gross (1999) the lexicographic data available in machine-readable format are printed dictionaries, electronic dictionaries and corpora. In particular dictionaries are built for being used by programs, with their content made of alphanumerical codes which represent the grammatical data that can be reasonably formalized at this moment in time. The creation and management of the electronic dictionary of MWEs in Italian spoken language took place through four main steps:

- lexical acquisition from corpus
- lexicon-based identification of MWEs
- information extraction
- identification of most recurrent PoS patterns

The first step concerns the *lexical acquisition*. We automatically extract MWEs starting from PoliModalCorpus (Trotta et al., 2018), a political domain corpus for Italian language currently composed of transcriptions[4] of 59 face-to-face interviews (14:00:00 hours) held during the political talk show "In mezz'ora in più" (from 24 September 2017 to 14 January 2018) and 18 speeches (7:02:39 hours) held during the election campaign for regional elections (from December 24th 2014 to March 4th 2015) by the then candidate Vincenzo De Luca[5]. The dimension of the individual corporus is indicated below (Tab. 1).

	Type	**Token**	**TTR**
PoliModalCorpus	11,231	158,543	0.07
De Luca Corpus	7,225	56,672	0.12
Total	18,456	215,251	0.08

Table 1 - Corpus statistics overview

In a second step – exploiting the theoretical backgroung offered by the Lexicon-Grammar[6] framework - we identified the MWEs by processing the corpus in Nooj[7] (Elia et al., 2010) and using the Compound-Word Electronic Dictionaries (DELAC-DELACF) (De Bueriis & Elia, 2008), which includes compound words and sequences formed by two or more words which jointly construct single units of meaning, thanks to which it was also possible to attribute a terminological label to each identified MWEs. It has to be noticed that in this step our efforts focused on the extraction of nominal compounds, leaving the extraction and integration of adverbial and adjectival compounds for future research. In a third phase the extracted MWEs were manually verified using the GRADIT (De Mauro, 2000). This operation has allowed us to identify 356 MWEs compared to 882 identified by DELAC-DELACF and to attribute to each compound expression the respective frequency label documented by the GRADIT. In a fourth phase a structural analysis of the extracted MWEs was carried out and the most recurring part of speech patterns were identified. Therefore the terminological labels[8] are distributed as follows: <econ> 112, <fig> 37, <dige> 36, <pol> 21, <med> 17[9]. Even though we extracted the MWEs from interviews of political kind, the MWEs tagged with the <pol> (political) labels are only 21. Following the most recurrent frequency label we found were: TS[10] (167) (i.e. *abuso di ufficio*), CO[11] (136) (i.e. *arredo urbano*), CO - TS (30) (i.e. *istituto di credito*). The methodological approach of the Lexicon-grammar has also restricted the taxonomic

[4] Using a semi-supervised speech-to-text methodology (Google API + manual transcription).

[5] It should be specified here that our is an initial experiment on a relatively small sample, since a larger balanced corpus would be necessary for a broader coverage.

[6] Gross (1975) shows that every verb has a unique behavior, characterized by different properties and constraints. In general, no ether verb has an identical syntactic paradigm. Consequently, the properties of each verbal construction must be represented in a lexicon-grammar.

[7] NooJ is a knowledge-based NLP tool based on huge hand-crafted linguistic resources, i.e. Dictionaries, derivational grammars. (Vietri, 2014).

[8] Being an essentially terminological dictionary, DELAC-DELACF assigns one or more terminology labels to each single entry, based on the areas of knowledge in which a specific compound has been attested. Currently the domains are 173 and the most populated is that of medicine.

[9] The terminological labels with a frequency lower than 17 are not mentioned.

[10] Technical-specialist use (107,194 words have this acronym and are known above all in relation to specific contexts of science or technology, eg amicina).

[11] Common use (as many as 47.060 words are used and understood and understood, regardless of profession or origin, to anyone with a higher level of education, eg allusivo).

analysis of compound polysematic words today they are naturally combined with the notion of compound nouns set by Gross and which can be described as "the sequence of their grammatical categories, in the same way as for adverbs" (Gross, 1986). Starting from this point of view, we may indicate how the most recurring patterns in our dictionary were respectively: *N + A* - valid for 218 words (like *lavori forzati* ecc), *N di N* (82) (i.e. *economia di scala*), *N + N* (30) (i.e. *estratto conto*), *N prep N* (22) (i.e. *ministero del lavoro*), *N a N* (2) (i.e. *corpo a corpo*), *N da N* (2) (i.e. *macchina da guerra*). Notice that, since in this study we are dealing with nominal MWEs the syntactic head of the compounds is always represented by the name in patterns like *N + A* and *A + N*, *N + N*, while in more complex patterns, as *N a N* and the like, we found controversial the identification of a single word as syntactic head. Since our primary interest was to identify and systematically arrange the extracted knowledge from a lexicographic point of view, we decided to deepen the syntactic analysis (which is to say the explicitation of the syntactic heads and the syntactic category of each MWE) during research steps to be included in near future research. Starting from the information extracted so far we have then created an electronic dictionary where to each MWE are associated information about gender and number, part of speech pattern, frequency labels, and terminological label. The dictionary was created using the XML as markup language following the TEI standard[12] and adding the tags `<mark>` in order to include the frequency tags indicated by the GRADIT and `<label>` to indicate the knowledge domain in which the word is attested, indicated to the DELAC-DELACF dictionaries). The choice of exploiting this markup language is motivated by its extreme generalization and flexibility (Pierazzo, 2005) and in order to represent the MWEs in a common format and to enable linkage (Calzolari et al., 2002). The adopted formalism uses the following tags:

- `<entry>`: contains a single structured entry in any kind of lexical resource, such as a dictionary or lexicon
- `<form>`: (form information group) groups all the information on the written and spoken forms of one headword
- `<gramGrp>`: (grammatical information group) groups morpho-syntactic information about a lexical item, e.g. pos, gen, number
- `<mark>`: frequency label from GRADIT
- `<label>`: terminological label from DELAC-DELACF

The dictionary therefore appears as follows:

```
<entry>
  <form>
    <orth>abuso d'ufficio</orth>
    <type>multiword expression</type>
 </form>
 <gramGrp>
  <gram type= "pos">NdiN</gram>
   <gram type="gen">m</gram>
   <gram type="num">s</gram>
 </gramGrp>
 <mark>TS</mark>
 <label>dige</label>
</entry>

<entry>
  <form>
    <orth>agente atmosferico</orth>
    <type>multiword expression</type>
 </form>
 <gramGrp>
  <gram type= "pos">NA</gram>
   <gram type="gen">m</gram>
   <gram type="num">s</gram>
 </gramGrp>
 <mark>TS</mark>
 <label>meteor</label>
</entry>
```

4 Ontologic expansion of the xml dictionary

Following the creation of the dictionary we also decided to organize the knowledge retrieved from the exploited datasets as an ontological dictionary which is actually under construction and that will be freely avilable under Creative Commons License (CC+BY-NC-ND). The choice to build such a linguistic resource is grounded on the idea that a formal representation of the MWEs may not only help software agents in the automatic recognition of compound words within written/oral texts, but can still enhance the resolution of referential expression such as *Primo Ministro*, *Santo Padre* and the like, which is to say of those frozen expressions that bear pragmatic references pointing to subject/object

[12] P5: Guidelines for Electronic Text Encoding and Interchange, Version 3.4.0. Last updated on 23rd July 2018, revision 1fa0b54.

that are likely to change over medium/short periods of time. In order to perform a deeper pragmatic disambiguation of MWEs we exploited the descriptive capability of the Ontology Web Language (OWL), a standard markup language provided by the World Wide Web (W3C) Consortium for the formalization of vocabularies of terms covering specific domains of knowledge. Following the W3C guidelines we shaped the electronic dictionary so that to each MWE a set of description classes and linking relationship are attached, according to the lexicon-grammar analysis previously performed and transposed into the ontology. Here is an example of the metadata scheme provided for the compound expression *campagna elettorale:*

- **Class "DELAC-DELACF Label"**: <pol> (politic)
- **Class "GRADIT" Label**: CO (Common)
- **Class "Syntactic Pattern"**: N(oun) + A(djective)
- **Data property "Corpus frequency"**: 52
- **Data property "Occurrence"**: *Berlusconi comincia la sua* ***campagna elettorale*** *andando in Tunisia a commemorare Craxi, che ne pensa di questa decisione?*
- **Data property "DBpedia redirection link"**: http://it.dbpedia.org/resource/Campagna_elettorale/html

As we can notice the first three classes plus the first two data properties directly derive from the linguistic analysis and their ontological formalisation may serve as powerful search filters in case of description logic queries submitted over the electronic dictionary. To what concerns the DBpedia redirection link property class, this derives from the Italian section of DBpedia project (Auer *et al.*, 2007) and will serve as core mechanism for the pragmatic resolution of the compound expression. It should be further noticed that the mapping effort between the extracted MWEs and DBpedia virtually put the work in progress ontology on the fifth and last level of Berner Lee's Open Data scale, which is to say on the level reserved for web semantic compliant resources additionally providing redirection links to other web datasets for the contextualisation of the described knowledge, following the initial proposal of (Bizer *et al.*, 2008).

5 Future work

In this work we described the initial steps for the development and formalization of PoliSdict, an electronic dictionary of spoken language MWEs. We illustrated the methdology used to build the resource and the preliminary results that we obtained from a systematic analysis. For what is related to future research we consider necessary exploiting standard association measures (like mutual information or log-likelihood ratio) to get an index of cohesion within the identified expressions and compare the use and collocations of MWEs between corpora of written and spoken language in order to understand which of them are the most used. Considering this study as an initial experiment on a relatively small sample, a larger balanced corpus would be necessary for a broader coverage, therefore we intend to proceed with the expansion of the corpus and the associated dictionary. Following we will make the described resources freely accessible by means of graphical interface, so as to offer the possibility to browse and explore data, also allowing the free use of the source codes for research purposes under Creative Commons License (CC+BY-NC-ND.

6 Aknowledgments

We would like to thank Network Contacts s.r.l. for their willingess to help us with valuable research insights and for the support during the writing of this paper. We would also like to thank the anonymous reviewers for their helpful suggestions.

References

Baldwin, T., & Villavicencio, A. (2002, August). Extracting the unextractable: A case study on verb-particles. In *proceedings of the 6th conference on Natural language learning-Volume 20* (pp. 1-7). Association for Computational Linguistics.

Bejček, E., & Straňák, P. (2010). Annotation of multiword expressions in the Prague dependency treebank. *Language Resources and Evaluation*, *44*(1-2), 7-21.

Bel, N., Poch, M., & Toral, A. (2012). *PANACEA (Platform for Automatic, Normalised Annotation and Cost-Effective Acquisition of Language Re-sources for Human Language Technologies)*. In

Proceedings of the 16th Annual Conference of the European Association for Machine Translation, Trento, Italy.

Calzolari, N., Fillmore, C. J., Grishman, R., Ide, N., Lenci, A., MacLeod, C., & Zampolli, A. (2002, May). Towards Best Practice for Multiword Expressions in Computational Lexicons. In LREC.

Chiari, I. (2012). Collocazioni e poliremathiche nel lessico musicale italiano. *Lingua, letteratura e cultura italiana". Atti del convegno Internazionale*, *50*, 165-190.

CSL. 2005. Special issue on Multiword Expressions of Computer Speech & Language, volume 19.

D'Agostino, E., & Elia, A. (1998). Il significato delle frasi: un continuum dalle frasi semplici alle forme poliremathiche. *AA. VV, Ai limiti del linguaggio. Bari: Laterza*, 287-310.

Dagan, I., & Church, K. (1994, October). Termight: Identifying and translating technical terminology. In *Proceedings of the fourth conference on Applied natural language processing* (pp. 34-40). Association for Computational Linguistics.

Daille, B. (1995). Combined approach for terminology extraction: lexical statistics and linguistic filtering.

De Bueriis, G., & Elia, A. (2008). Lessici elettronici e descrizioni lessicali, sintattiche, morfologiche ed ortografiche. *Plectica, Salerno*.

De Mauro, T. (1999). Gradit. *Torino: UTET*, *1*.

Maienborn, C., von Heusinger, K., & Portner, P. (Eds.). (2011). *Semantics: An international handbook of natural language meaning* (Vol. 1). Walter de Gruyter.

Grégoire, N. (2010). DuELME: a Dutch electronic lexicon of multiword expressions. *Language Resources and Evaluation*, *44*(1-2), 23-39.

Gross, G. (2018). Thématisation des compléments circonstanciels. In *Le poids des mots. Hommage à Alicja Kacprzak;*. Wydawnictwo Uniwersytetu Łódzkiego.

Gross, M. (1986, August). Lexicon-grammar: the representation of compound words. In *Proceedings of the 11th conference on Computational linguistics* (pp. 1-6). Association for Computational Linguistics.

Gross, M. (1999). A bootstrap method for constructing local grammars. In *Proceedings of the Symposium on Contemporary Mathematics* (pp. 229-250). University of Belgrad.

JLRE. 2009. Special issue on Multiword Expressions of the Journal of Language Resources and Evaluation, volume to appear.

Kim, S. N., & Baldwin, T. (2006, April). Automatic identification of English verb particle constructions using linguistic features. In *Proceedings of the Third ACL-SIGSEM Workshop on Prepositions* (pp. 65-72). Association for Computational Linguistics.

Kim, S. N., & Baldwin, T. (2006, April). Automatic identification of English verb particle constructions using linguistic features. In *Proceedings of the Third ACL-SIGSEM Workshop on Prepositions* (pp. 65-72). Association for Computational Linguistics.

McEnery, T., Langé, J. M., Oakes, M., & Véronis, J. (1997). The exploitation of multilingual annotated corpora for term extraction. *Corpus annotation---linguistic information from computer text corpora*, 220-230.

Michiels, A., & Dufour, N. (1998). DEFI, a tool for automatic multi-word unit recognition, meaning assignment and translation selection. In *Proceedings of the first international conference on language resources & evaluation* (pp. 1179-1186).

Monti J., di Buono M.P., Sangati, F. (2017) PARSEME-It Corpus *An annotated Corpus of Verbal Multiword Expressions in Italian*. In: CLIC-It 2017 Proceedings - Rome 11-13 December 2017.

Mudraya, O., Babych, B., Piao, S., Rayson, P., & Wilson, A. (2006). Developing a Russian semantic tagger for automatic semantic annotation. *Corpus Linguistics 2006*, 290-297.

Nunberg, G., Sag, I. A., & Wasow, T. (1994). Idioms. *Language*, *70*(3), 491-538.

Pearce, D. (2002, May). A Comparative Evaluation of Collocation Extraction Techniques. In *LREC*.

Pecina, P. (2010). Lexical association measures and collocation extraction. *Language resources and evaluation*, *44*(1-2), 137-158.

Piao, S., Archer, D., Mudraya, O., Rayson, P., Garside, R., McEnery, T., & Wilson, A. (2005). A large semantic lexicon for corpus annotation. *Corpus Linguistics 2005*.

Pierazzo, E. (2005). *La codifica dei testi: un'introduzione*. Carocci editore.

Ramisch, C., Schreiner, P., Idiart, M., & Villavicencio, A. (2008, June). An evaluation of methods for the extraction of multiword expressions. In *Proceedings of the LREC Workshop-Towards a Shared Task for Multiword Expressions (MWE 2008)* (pp. 50-53).

Rayson, P., Archer, D., Piao, S., & McEnery, A. M. (2004). The UCREL semantic analysis system.

Smadja, F. (1993). Retrieving collocations from text: Xtract. *Computational linguistics*, *19*(1), 143-177.

Strik, H., & Cucchiarini, C. (1999). Modeling pronunciation variation for ASR: A survey of the literature. *Speech Communication*, *29*(2-4), 225-246.

Strik, H., Hulsbosch, M., & Cucchiarini, C. (2010). Analyzing and identifying multiword expressions in spoken language. *Language resources and evaluation*, *44*(1-2), 41-58.

Trotta, D., Albanese, T., Elia, A., *Polimodalcorpus: verso la costruzione del primo corpus multimodale di dominio politico in italiano;* Proceedings of the XXVIII Ass.I.Term International Conference, Salerno, 2018.

Vietri, S. (2004). *Lessico-grammatica dell'italiano. Metodi, descrizioni e applicazioni* (p. 304). UTET Università.

Vietri, S. (1985). *Lessico e sintassi delle espressioni idiomatiche: una tipologia tassonomica dell'italiano*. Liguori.

Vietri, S. (2014). *Idiomatic constructions in Italian: a lexicon-grammar approach* (Vol. 31). John Benjamins Publishing Company.

Villavicencio, A., Kordoni, V., Zhang, Y., Idiart, M., & Ramisch, C. (2007). Validation and evaluation of automatically acquired multiword expressions for grammar engineering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Wermter, S., & Chen, J. (1997). Cautious steps towards hybrid connectionist bilingual phrase alignment. In *Recent Advances in Natural Language Processing* (Vol. 97).

Wu, D. (1997). Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational linguistics*, *23*(3), 377-403.

Zhang, Y., Kordoni, V., Villavicencio, A., & Idiart, M. (2006, July). Automated multiword expression prediction for grammar engineering. In *Proceedings of the workshop on multiword expressions: Identifying and exploiting underlying properties* (pp. 36-44). Association for Computational Linguistics.