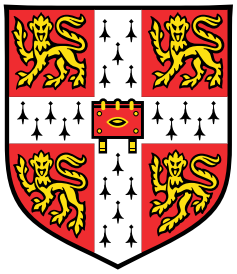
# Representation Learning beyond Semantic Similarity: Character-aware and Function-specific Approaches



**Daniela S. Gerz**

Department of Theoretical and Applied Linguistics
Language Technology Lab (LTL)
University of Cambridge

This dissertation is submitted for the degree of
*Doctor of Philosophy*

Lucy Cavendish College

April 2020

# Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 80,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Daniela S. Gerz
April 2020

# Acknowledgements

Completing this dissertation would not have been possible without the help of all the incredible people I was fortunate to meet during my studies. Thank you Ivan for all the mentoring, patience and brilliant advice. Thank you for always taking time, caring, supporting and backing me. I learned so much from your guidance and continuous feedback. Thank you Anna for believing in me, and for being the most understanding and supportive advisor I could have wished for. Roi, thank you for teaching me the value of thorough scientific work. I got a lot more out of my time in Cambridge than I had dared to dream. Thanks to all of you for making this possible.

Thank you to all friends and fellow NLP people in LTL, the computer lab, engineering, in London and around the world. Thank you Gamal, Billy, Milan for figuring out Cambridge together, as well as Edoardo, Olga, Jason, SangSeo, Ehsan, Costanza, Yan, Flora and all of LTL for keeping me company in the Brown Library and beyond. Thank you Eddy, Shawn and Nikola for your patience and supporting me writing up. Cheers to Marek, Kris, and everyone in PolyAI (please do feel named personally) for all the good company, parties and adventures around the world. This thesis would not exist if it wasn't for all the much welcome breaks with you.

Thank you Lissie, Lizzie, Kirstyn and Florence for helping me through an especially challenging first year. Thank you Saakshi, Issy, Shraddha, and all of 109. The year with you has made me realise so many things and genuinely changed my life for the better.

Finally, I want to thank my parents, grandparents and family back home. Thank you for your unconditional love and support. Thank you for inspiring me to be courageous, for always believing and backing me. To my dear mom, who saw the beginning of this dissertation but unfortunately is not around anymore to witness its end. Thank you for being the most caring and kind person possible, for always being by my side and encouraging me to follow my own path. Papa, ich danke dir von ganzem Herzen. For always being there for me, reminding me to be strong and not afraid of failure. Because especially in the most dire situations, a reminder to keep calm and happy can take you incredibly far.

# Abstract

Representation learning is a research area within machine learning and natural language processing (NLP) concerned with building machine-understandable representations of discrete units of text. Continuous representations are at the core of modern machine learning applications, and representation learning has thereby become one of the central research areas in NLP. The induction of text representations is typically based on the distributional hypothesis, and consequently encodes general information about word similarity. Words or phrases with similar meaning obtain similar representations in a vector space constructed for this purpose. This established methodology excels for morphologically-simple languages such as English, and in data-rich settings. However, several useful lexical relations such as entailment or selectional preference, are not captured or get conflated with other relations. Another challenge is dealing with low-data regimes for morphologically-complex and under-resourced languages. In this thesis we construct novel representation learning methods that go beyond the limitations of the distributional hypothesis and investigate solutions that induce vector spaces with diverse properties. In particular, we look at how the vector space induction process influences the contained information, and how the information manifests in a number of core NLP tasks: semantic similarity, lexical entailment, selectional preference, and language modeling. We contribute novel evaluations of state-of-the-art models highlighting their current capabilities and limitations. An analysis of language modeling in 50 typologically-diverse languages demonstrates that representations can indeed pose a performance bottleneck. We introduce a novel approach to leveraging subword-level information in word representations: our solution lifts this bottleneck in low-resource scenarios. Finally, we introduce a novel paradigm of function-specific representation learning that aims to integrate fine-grained semantic relations and real-world knowledge into the word vector spaces. We hope this thesis can serve as a valuable overview on word representations, and inspire future work in modeling *semantic similarity and beyond*.

# Contents

# List of Abbreviations

- **AP** Attract-Preserve
- **AS** Additional Supervision
- **BBC** British Broadcasting Corporation
- **BNC** British National Corpus
- **BOW** Bag of words
- **BWB** 1 Billion Word Benchmark
- **CF** Counter-Fitting
- **Char** Character
- **CV** Cross-Validation
- **CNN** Convolutional Neural Network
- **DEM** Directional Entailment Measures
- **DEPS** Dependency Contexts
- **FR** Concept Word Frequency
- **HYP** Hyper/Hyponym
- **IAA** Inter-Annotator Agreement
- **IMDB** Internet Movie Database
- **LE** Lexical Entailment
- **LM** Language Modelling
- **LSTM** Long short-term memory
- **KN** Kneser-Ney
- **MRL** Morphologically-Rich Languages
- **MWC** Multilingual Wikipedia Corpus

- **NLP** Natural Language Processing
- **NLU** Natural Language Understanding
- **NR** No Relation
- **OOV** Out-of-Vocabulary
- **PA** Prolific Academic
- **PMI** Pointwise Mutual Information
- **PTB** Penn Treebank
- **PPDB** The Paraphrase Database
- **PPL** Perplexity
- **PW** Polyglot Wikipedia
- **ReLU** Recurrent Linear Unit
- **RNN** Recurrent Neural Network
- **SDF** Sparse Distributional Features
- **SDSN** Supervised Directional Similarity Network
- **SGD** Stochastic Gradient Descent
- **SGNS** Skip-Gram Negative Sampling
- **SL** SimLex
- **SMT** Statistical Machine Translation
- **SV** Subject Verb
- **SVO** Subject Verb Object
- **SYN** Synonym
- **TTR** Type-Token Ratio
- **UNK** Unknown Word
- **USF** University of Southern Florida

- **V** Vocabulary
- **VC** Verb Class
- **VN** VerbNet
- **VO** Verb Object
- **VSM** Vector Space Model
- **WN** WordNet

# List of Figures

# List of Tables

# Chapter 1

# Introduction

> The limits of my language mean the limits of my world.
>
> *Ludwig Wittgenstein*

## 1.1 Motivation

Representation learning is a key research area within machine learning and natural language processing (NLP) concerned with building machine-understandable representations of discrete units of text. Continuous representations form one of the most fundamental parts of virtually every modern machine learning approach in NLP (Collobert and Weston, 2008; Collobert et al., 2011; Bengio et al., 2003). Applications as diverse as machine translation (Cho et al., 2014; Sutskever et al., 2014; Mikolov et al., 2010), parsing (Chen and Manning, 2014) and response selection (Cer et al., 2018; Yang et al., 2018; Henderson et al., 2017) all make use of representations for basic natural language understanding (NLU).

A classic way to represent words are *one-hot* vectors, where each vector dimension represents one word in the vocabulary. In this approach, all dimensions in the vector are set to zero, and only one dimension representing the current word is assigned the value *one* (Figure 1.1a). This not only leads to extremely large vectors (growing with the size of the vocabulary) and expensive computation, but also does not grant a model much flexibility to learn about relations. In a vocabulary of thousands of words, related concepts such as *hotel* and *hostel* might get assigned random dimensions, and consequently the model is unable to learn any connection between them.

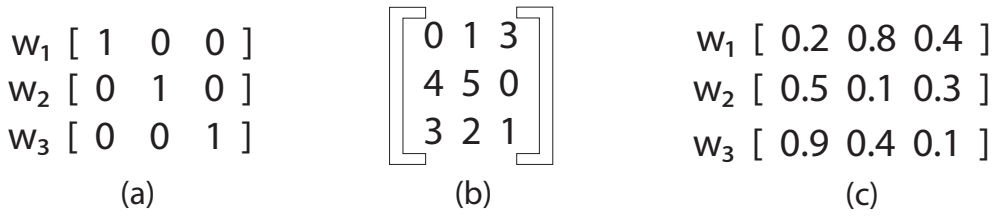



Figure 1.1 Approaches to vector space induction. **(a) One-hot** vectors. **(b) Co-occurrence** (word-context frequency) matrix. **(c) Dense** vectors.

One way to mitigate this issue is to learn text representations on the basis of the distributional hypothesis, which states that words occurring in similar contexts will have a similar meaning (Harris, 1954; Firth, 1968). Representations built on this assumption typically look at word-context co-occurrences in a large corpus and thereby can learn to encode general word similarity (Hill et al., 2015). Earlier count-based models generally were operating in a two-step approach: (1) counting all word-context co-occurrences from a corpus and collecting them in a matrix (Figure 1.1b), then (2) applying dimensionality reduction techniques to compress the counts into dense vectors (Levy et al., 2015a; Turney et al., 2010; Hofmann, 1999; Landauer et al., 1998; Deerwester et al., 1990, inter alia). These models typically represent each word with one dense vector (Figure 1.1c), and words or phrases with similar meaning will be represented with vectors lying close together in vector space (Schütze, 1993). For example, we would expect similar words such as *hotel* and *hostel* to be nearest neighbours. Most recent techniques directly induce the lower dimensional dense vector (Levy et al., 2015a) by predicting words in context (Mikolov et al., 2013a,b; Pennington et al., 2014; Bojanowski et al., 2017, *inter alia*). While the meaning of single dimensions in dense vectors remains unclear[1], typically the number of dimensions can simply be fixed to a small integer, making computation vastly more efficient compared to prior approaches (Mikolov et al., 2013a). Thanks to this increased efficiency and the many conceptual and practical advantages coming with it, word representations have found broad application across the field of NLP. In particular, computationally efficient word representations can be trained in an unsupervised way from large corpora, leveraging the large amounts of raw text increasingly available online (Al-Rfou et al., 2013; Henderson et al., 2019). Representations trained from such data sets can therefore contribute information about general word meaning, which is especially helpful in supporting task-specific models, as task-specific data typically is expensive to obtain and therefore lacks broad coverage of general language usage.

However, there remain several limitations that are yet to overcome. With the training objective of predicting any word in a context window, word representations inherently are

[1]The lack of understanding that comes with dense vectors in neural networks has even given rise to a workshop series: BlackboxNLP.

agnostic to word order and syntactic patterns.[2] We observe that with one vector per word to represent all its meanings, many representational models conflate multiple semantic relations (Mrkšić et al., 2017). In addition, a focus on word-level processing is appropriate for morphologically-simple languages such as English in data-rich settings, but not a very suitable solution when faced with data sparsity issues occurring especially in morphologically-rich languages (Part III). Hence a crucial challenge to tackle thereby is going beyond the limitations of purely context window based distributional training, and expanding research on representation learning into two complementary dimensions 1) broader coverage of different semantic relations and linguistic phenomena 2) a more diverse set of languages, with a focus on morphologically-rich languages where awareness of subword information can be of tremendous help.

Let us consider a real-world task to illustrate the limitations of general word embeddings. For instance, a dialogue system for restaurants should be able to take reservations as well as answer questions about the menu offering of the restaurant. Quite a few pieces of information are relevant to complete this task successfully, as illustrated on a toy example in Figure 1.2.

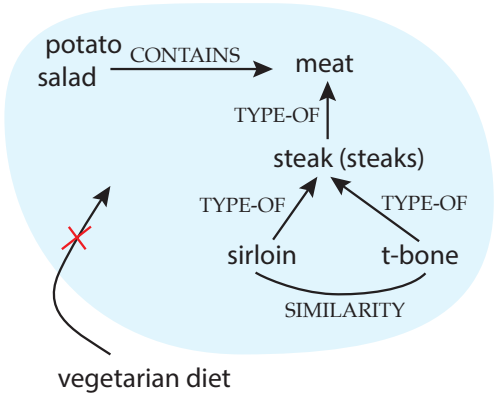



Figure 1.2 Toy example illustrating the interplay of lexical relations and real-world knowledge an automatic QA system might need to reason over.

The system needs knowledge of the dishes on the menu (e.g. *6oz sirloin steak)* and its ingredients (e.g. *meat, potatoes*). Furthermore if someone asks for a steak, it should know that *sirloin* or *t-bone* are types of *steak* (**Semantic Similarity**) and types of *meat* (**Lexical Entailment**). Equally, if someone says they are vegetarian it should know that vegetarians do not eat meat and exclude the steaks as well as any other type of meat from the

[2]While a recent strand of *context-aware* models addresses this point (Peters et al., 2018; Devlin et al., 2019), further limitations persist. In future work we discuss how the findings of this thesis might be integrated with these newer models.

recommendations altogether (**Association / Real-world knowledge**). Finally, the capability of the system to perform well on these tasks should be agnostic to morphology. In English that means the system should recognise both singular and plural forms (*steak*, *steaks*). For morphologically-rich languages, there may be a significant number of forms to cover for each word.

All of the above-mentioned information is important to solve the task correctly, as otherwise the system might give the wrong response. If the system is missing the ability to reason over semantic similarity, it can do little more than keyword matching when the exact word has not been observed previously. If knowledge about entailment is absent, the system may not be able to find *steak* if the user is asking for *meat*. If knowledge about morphology is absent, the system may not be able to find *steak* but not *steaks*. If associations or real-world knowledge is absent, it will not know that *potato salad* often contains *meat stock*, which is unsuitable for *vegetarians*. As a bottom line, we need to be aware of what we are encoding, what we want to encode, and how that can enhance or place a limit on our systems.

This insight leads us to explore representation learning *beyond semantic similarity* in this dissertation. We consider a complementary range of methods for encoding information with artificial neural networks, and contribute especially to previously under-resourced phenomena. In particular, our work aims at a broader coverage of semantic relations and a larger variety of languages.

The next sections give an overview of the dissertation. We then proceed to introduce key topics before delving into the three main parts: semantic relations, language modeling for morphologically-rich languages, and function-specific word representations. We hope this dissertation can serve as a valuable contribution to representation learning, and inspire further research into modeling semantic similarity and beyond.

## 1.2 Research Objectives

The basic premise of this dissertation is the ideology that a set of task and language independent representations of language can be used universally across the field of NLP as a base layer for natural language understanding (NLU). However, many semantic relations may not get covered or get conflated into general distributional vectors. Furthermore, there exists an implicit bias towards morphologically-simple languages (such as English), which leads to considerably lower performance in morphologically-rich languages. We hypothesise that for general, language-independent NLU more fine-grained coverage of semantic relations and morphology will be necessary. In particular we consider the following desiderata:

- **Wide-coverage evaluation for semantic similarity** Semantic similarity and relatedness are the most commonly evaluated relations. We want to ensure that evaluation sets provide a good coverage of linguistic phenomena, spanning a variety of word types and frequencies.
- **Awareness of the influence of typological factors on model performance** A key challenge is developing language-independent architectures that can be successfully applied across a wide range of languages. To this end we want to ensure awareness of how typological factors influence model performance.
- **Modeling for morphologically-rich languages** Morphologically-rich languages can be challenging to model with standard word-level methods, due to data sparsity issues arising from a large number of infrequent words. We want to create modeling approaches and architectures that can work across the full spectrum of the world's languages, including morphologically-rich languages.
- **Representations beyond semantic similarity** We want to look at how to evaluate and model representations for relations beyond semantic similarity, and ensure that both relation-specific and domain-specific knowledge can be adequately represented.

## 1.3 Contributions

This disseration summarises a number of contributions across the field of representation learning. We contribute towards under-resourced phenomena, in particular considering different semantic relations as well as a broader linguistic diversity.

We contribute to evaluation benchmarks that enable evaluation for particular semantic relations such as true semantic similarity or lexical entailment (SimVerb-3500, HyperLex-2616), new models and modeling paradigms (character-aware next-word prediction, function-specific learning), and benchmark language models across a typologically-diverse set of 50 languages. More specifically:

- We introduce a novel resource for *verb similarity* analysis and evaluation, **SimVerb-3500**. The data set provides human ratings for 3,500 English verb pairs, which is significantly larger than previous resources. It also assures a broad coverage of syntactic and semantic phenomena, which makes it possible to compare the strengths and weaknesses of various representation models via statistically robust analyses on specific word classes.

- We contribute a data set for lexical entailment, **HyperLex** (Vulić et al., 2017). Similar to Simlex-999 and SimVerb-3500, HyperLex provides graded scores of word pairs. We provide a benchmark of existing modeling approaches for hyponymy-hypernymy, a comparison of similarity vs lexical entailment, as well as a modeling approach that can map distributional vector space to entailment (Rei et al., 2018).

- We provide an analysis on the relation of word representations to linguistic typology by conducting a large-scale benchmark of language modeling (LM), surveying a range of state-of-the-art LM architectures across a **typologically-diverse set of 50 languages**. We conclude that in fact even "open-vocabulary" architectures often contain a bottleneck by virtue of their word representations. We detect a correlation to selected typological features to the level of LM performance, which demonstrates that morphologically-rich languages are especially hard to model with current architectures.

- We attempt to lift the bottleneck in LM by introducing a novel training paradigm: **character-aware next-word prediction**. Especially for morphologically-rich languages a lot of semantic information is contained in subword features. A popular language model (CharCNN-LSTM) learns subword features in one of its input layers, but does not propagate this information to its word representations in the output layer.

We periodically inject these model-internal subword features into its word representations while training via an additional training objective. This improves performance especially in morphologically-rich languages, which otherwise suffer from extreme data sparsity issues.

- Finally, we introduce a novel paradigm for creating word representations: **function-specific modeling**. Our joint model learns interrelated representations of disjoint vocabularies, such as the ones found in Subject-Verb-Object (SVO) structures. We find the resulting vectors are effective on a number of tasks which reason over the SVO structure. The vectors reach or surpass state-of-the-art performance on estimating selectional preference (thematic-fit) and event similarity, which previously has only been possible with more complex, task-specific models. Our approach outperforms these task-specific architectures while reducing the number of parameters by up to 95%. The resulting model can be applied to create vector representations for many semantic and syntactic phenomena. In future work we hope it can also be used to create representations for factual or real-world knowledge, thus making this information differentiable and thereby accessible to neural architectures.

## 1.4 Publications

The material discussed in this dissertation mainly relates to the following articles, in order of publication:

1. **Daniela Gerz**, Ivan Vulić, Felix Hill, Roi Reichart and Anna Korhonen. "SimVerb-3500: A large-scale evaluation set of verb similarity." *EMNLP 2016*.

2. Ivan Vulić, **Daniela Gerz**, Douwe Kiela, Felix Hill, and Anna Korhonen. "Hyperlex: A large-scale evaluation of graded lexical entailment." *Computational Linguistics*, 43(4). *2017*.

3. Marek Rei, **Daniela Gerz**, and Ivan Vulić. "Scoring Lexical Entailment with a Supervised Directional Similarity Network." *ACL 2018*.

4. **Daniela Gerz**, Ivan Vulić, Edoardo Maria Ponti, Roi Reichart, and Anna Korhonen. "On the Relation between Linguistic Typology and (Limitations of) Multilingual Language Modeling." *EMNLP 2018*.

5. **Daniela Gerz**, Ivan Vulić, Edoardo Maria Ponti, Jason Naradowsky, Roi Reichart, and Anna Korhonen. "Language modeling for morphologically rich languages: Character-

aware modeling for word-level prediction." *Transactions of the Association of Computational Linguistics 6, 451-465. 2018.*

6. Ehsan Shareghi, **Daniela Gerz**, Ivan Vulić, Anna Korhonen. "Show Some Love to Your *n*-grams: A Bit of Progress and Stronger *n*-gram Language Modeling Baselines." *NAACL-HLT 2019*

7. **Daniela Gerz**, Ivan Vulić, Marek Rei, Roi Reichart, and Anna Korhonen. "Associative Learning for Function-Specific Word Representations." (Long paper under review)

## 1.5 Thesis Overview

This dissertation is structured into four main parts. Part I introduces the main topics and tasks we will work on in later sections. Parts II, III and IV each deal with different kinds of representations.

**Part II: Semantic Relations** focuses on word-level semantic relations. We contribute two word-level intrinsic evaluation sets. Chapter 3 contributes a data set with human annotations for semantic similarity of verbs. Chapter 4 looks at graded lexical entailment. It introduces a data set, then discusses and evaluates modeling approaches to reason over graded lexical entailment.

**Part III: Language Modeling for Morphologically-Rich Languages** contains a benchmark spanning a range of language models across a typologically-diverse set of 50 languages. We find that using word representations to model morphologically-rich languages can be challenging due to typological differences and data sparsity. This part introduces character-aware next-word prediction, which injects subword-level knowledge into word representations to mitigate data sparsity.

**Part IV: Function-specific Word Representations** contributes a new modeling paradigm, used to create word representations specific to selected linguistic groups or structures. We demonstrate the efficiency of the model for the *subject-verb-object* structure, and suggest the same approach can be used to model a diverse range of phenomena.

# Part I

# Background

# Chapter 2

# Key Topics

> You shall know a word by the company it keeps
>
> John Rupert Firth

This chapter introduces key topics and tasks relevant to this dissertation. We give a high-level overview on a range of evaluations aimed at estimating the quality of learned representations. Further background and related work of individual topics will be provided in the subsequent chapters.

Estimating the quality of learned representations is not straightforward. As representations are typically learned in an unsupervised fashion from large corpora, it is not reliable to estimate the model quality directly from its predictions. Training objectives will vary across architectures, and usually do not directly correspond to the task the model will be applied on. Therefore instead, secondary tasks are being used to systematically probe the ability of a model to represent aspects relevant to its desired application.

Typically, intrinsic evaluation tasks will measure the correlation of model predictions to scores given by human annotators. This methodology is well-established for the relation of semantic similarity as well as selectional preference (Section 2.1). As we will show in Chapter 4, the same methodology can be applied to other semantic relations. In the following we will introduce and give an overview on the semantic relations *similarity* and *relatedness* (Chapter 3), *entailment* (Chapter 4), as well as *selectional preference* and *event similarity* (Chapter 6). Further, as it is challenging to find a comparable data set spanning a large number of typologically-diverse languages, we will look at the task of language modeling (Chapter 5) as a proxy to analyse the influence of linguistic diversity.

## 2.1 Evaluating Word Representations

One natural way to evaluate the quality of representations is by judging the *similarity* of representations assigned to similar words. The vast majority of word representations is trained based on the Distributional Hypothesis (Harris, 1954; Firth, 1968; Sahlgren, 2008), which has given rise to the development of numerous models providing *distributional word vector spaces* (Mikolov et al., 2013a,b; Pennington et al., 2014; Bojanowski et al., 2017). The Distributional Hypothesis is characterised by the idea that *"you shall know a word by the company it keeps"* (Firth, 1968), i.e. words with similar meaning are assumed to occur in the same context. Consequently, a vector space build on this assumption is largely governed by general *similarity*, i.e. words with similar meaning lie close in this vector space.

**Extrinsic vs Intrinsic Evaluation** Accurate measures of judging the quality of word vector representations is a frequent topic of discussion;[3] undoubtedly, transferring the pre-trained word vectors to larger models, using them for several *extrinsic* tasks is one method of evaluation. By using the vectors in various extrinsic tasks, we can directly measure their influence and application in practice. However, for several reasons, using an extrinsic evaluation may not always be feasible. Mainly, it can take up a lot of time and computational resources to set up, train and evaluate several state-of-the-art models and tasks. *Intrinsic* evaluations can give guidance in this situation without taking up enormous amounts of resources. Intrinsic evaluations typically consist of word pairs or triplets scored by human annotators according to given criteria. For example, in the case of distributional vector, one might ask annotators *"How similar do you think these two words are on a scale from 1 to 6?"* (Hill et al., 2015; Gerz et al., 2016).

### 2.1.1 Relatedness and Word Association

In some existing evaluation sets, such as RG-65 (Rubenstein and Goodenough, 1965) or WordSim-353 (Finkelstein et al., 2002), pairs are scored for *relatedness*, also called *word association*. Relatedness is a high-level lexical relation that subsumes many other lexical relations. Word association datasets such as the University of Southern Florida (USF) dataset (Nelson et al., 2004) are generated by giving participants a stimulus word, and then noting the *the first word that comes to their mind*. In other words, *two words are related if someone recalls one word given the other*. Consider for instance the two words *to run* and *to sweat*. The two words might occur in the same context, but are not interchangeable and carry distinct semantic meaning (McRae et al., 2012; Plaut, 1995; Hill et al., 2015). In the same way, one

[3] In fact it has even inspired a workshop series: Evaluating Vector Space Representations for NLP (RepEval)

might associate other words with running, such as an occasion (*marathon*) or items you would use (*shoes, towel*), that however all have a very distinct semantic meaning. Relatedness thereby can not be seen as informative for *true* semantic similarity.

### 2.1.2 Semantic Similarity

**Definition** Semantic Similarity in this thesis refers to a graded version of *synonymy*. According to the Oxford Dictionary[4] synonymy is *A word or phrase that means exactly or nearly the same as another word or phrase in the same language*. With semantic similarity here we refer to a measure of *how close two words are to being synonymous*. In other words, two words are considered similar if they have a highly similar semantic meaning. For example, *to run* and *to jog* can be considered semantically similar, since they represent similar actions.

**Similarity vs. Relatedness in Evaluation** In this dissertation we largely target *semantic similarity* as it is a more focused relation and has been shown to lead to higher correlation with downstream tasks (Chiu et al., 2016). A comparison of extrinsic vs intrinsic evaluations for distributional vectors is provided by Schnabel et al. (2015); Tsvetkov et al. (2015); Chiu et al. (2016). In particular, Chiu et al. (2016) highlight the best correlation to extrinsic tasks is achieved when the intrinsic evaluation explicitly differentiates true *semantic similarity* and *relatedness*. [5].

**The Special Case of Antonyms** One common disparity in evaluation sets are antonyms. Antonyms are rated low in SimLex-999 (Hill et al., 2015) and SimVerb-3500 (Gerz et al., 2016), but rated high in other data sets high due to their high relatedness. A one dimensional scale for rating antonymys therefore cannot fully capture their semantic properties, especially given that their treatment can depend on the requirements of the downstream task. For example in the case of sentiment analysis, antonyms such as *good* or *bad* should lead to different classification results. We make the assumption that for the majority of downstream tasks, modeling antonyms as dissimilar is more beneficial.

**Example Ratings** We now show some exemplary pairs rated for semantic similarity. The ratings in table 2.1 are taken from the SimVerb-3500 data set, which has been published as Gerz et al. (2016) and will be introduced in Chapter 3. These ratings were obtained in a similar way to other data sets, in that they consist of averaged ratings based on multiple human annotators. For instance *to repair* and *to fix* are considered highly similar by most

[4] https://www.lexico.com/en/definition/synonym

[5] A broader discussion on semantic similarity vs relatedness is provided by Hill et al. (2015)

| Pair | Rating |
|---|---|
| to repair / to fix | 9.96 |
| to instruct / to teach | 8.80 |
| to win / to achieve | 7.80 |
| to originate / to create | 5.64 |
| to wash / to spray | 4.65 |
| to entertain / to enjoy | 2.32 |
| to remove / to add | 0.17 |
| to visit / to giggle | 0.00 |

Table 2.1 Example verb pairs from SimVerb-3500 (Gerz et al., 2016).

human annotators and following that received an overall high score of *9.96*. Dissimilar pairs such as *to visit* and *to giggle* on the other hand received a low score by all annotators, similar to antonyms such as *to remove* and *to add*. Note that scores in the middle might either be a result of moderate similarity, or ambiguous i.e. context-dependent meanings: *to originate* and *to create* with a rating of *5.64* were rated high by some annotators, and low by others.

**Existing Evaluation Sets** A number of word pair evaluation sets are prominent in the distributional semantics literature for English. Representative examples include RG-65 (Rubenstein and Goodenough, 1965) and WordSim-353 (Finkelstein et al., 2002; Agirre et al., 2009) which are small (65 and 353 word pairs, respectively). Larger evaluation sets such as the Rare Words evaluation set (Luong et al., 2013) (2034 word pairs) and the evaluations sets from Silberer and Lapata (2014) are dominated by noun pairs and the former also focuses on low-frequency phenomena. These data sets do not provide a representative sample of verbs (Hill et al., 2015). Two data sets that do focus on verb pairs to some extent are the data sets of Baker et al. (2014) and Simlex-999 (Hill et al., 2015). These datasets, however, still contain a limited number of verb pairs (134 and 222, respectively), making them unrepresentative of the rich variety of verb semantic phenomena. For this reason we contributed a data set for verb similarity, which we introduce further in chapter 3.

### 2.1.3 Lexical Entailment

Another semantic relation that we will discuss further in Chapter 4 is lexical entailment (LE). Lexical entailment has the TYPE-OF or **hyponymy–hypernymy** relation at its core and occurs between category concepts and their constituent members (Vulić et al., 2017). For example, *orange* is a TYPE-OF *fruit*, *steak* is a TYPE-OF *meat*, or *dog* is a TYPE-OF *animal*. Entailment can be hierarchical as well, such as in *sirloin* is a TYPE-OF *steak* is a TYPE-OF *meat*, or *Golden Retriever* is a TYPE-OF *dog* is a TYPE-OF *animal*. Awareness of

such relations between concepts and categories is necessary, since humans intuitively reason about them (Quillian, 1967; Collins and Quillian, 1969). For example, a conversational agent that hears a user talking about their *Golden Retriever*, should be able to infer that the user is in fact talking about their dog and answer accordingly.

Traditionally lexical entailment has been an area of much focus in NLP, but has largely been treated as binary (Bos and Markert, 2005; Dagan et al., 2006; Baroni et al., 2012; Beltagy et al., 2013, inter alia). The relation between two words is thereby either declared as hypo/hypernym, or neither of the two. However, in many cases a decision is not clear cut; most people would agree that *dog* is a TYPE-OF *animal*, but for less prototypical examples such as *dinosaur*, *human being* or *amoeba* annotators' opinions might vary. **Graded lexical entailment** is tackling this issue by rating category membership on a scale, resulting in averaged scores similar to the ones used in semantic similarity data sets (Vulić et al., 2017).

### 2.1.4 Selectional Preference and Event Similarity

Another central semantic element is the **Subject-Verb-Object** (SVO) structure. The SVO structure can be seen as one of the most fundamental meaning-containing parts of a sentence. Even in the absence of all other parts of speech and inflection or declension, often the general meaning of a sentence can be read from a raw SVO structure alone. For instance, the following SVO triples all convey meaning even in the absence of other sentence parts: (*cat(S) - eat(V) - food(O)*), or (*researcher(S) - study(V) - science(O)*). Further, a statistical model for the SVO structure can also be seen as containing real-world knowledge and plausibility to some extent. For example, objects that go with the verb *to eat* can tell us about things that can be eaten. In combination with different subjects, it requires knowledge of diets and food intake. For example, the object predictions for *human(S) - eat (V)* should be different from *cat(S) - eat (V)*. The SVO structure has been researched in a number of different NLP tasks which reason about it from slightly different angles. We will elaborate on Selectional Preference modeling as well as Event Similarity in the following.

**Selectional Preference (SP)** also called *thematic-fit* or *plausibility* scoring, focuses primarily on verbs and their preferences for nouns. The task here is to accurately quantify whether a verb can take a certain subject (also called *agent*) or object (also called *patient*) (McRae et al., 1997; Sayeed et al., 2016). Selectional preference scoring is different from event similarity in that it is concerned with scoring word *pair* combinations, where one word is a verb and one is a noun, i.e. either *subject/verb* (SV) or *verb/object* (VO).

There are a number of standard benchmarks available which score selectional preference in a similar way. **1) MST1444** (McRae et al., 1998) contains 1,444 word pairs where humans

| Pair | Agent (SV) | Patient (VO) |
|---|---|---|
| **MST1444** | | |
| draw artist | 6.6 | 2.1 |
| enslave pirates | 6.3 | 2.7 |
| study scientist | 6.6 | 2.4 |
| **PADO414** | | |
| confuse computer | 5.4 | 4.1 |
| hear voice | 1.2 | 6.5 |
| promise sun-god | 1.2 | 2.7 |

Table 2.2 Example (verb, noun) pairs and respective agent (SV) and patient (VO) scores in MST1444 and PADO414 thematic fit data sets.

provided thematic fit ratings on a scale from 1 to 7 for each noun to score the plausibility of the noun taking the agent role, and also taking the patient role.[6] **2) PADO414** (Padó, 2007) is similar to MST1444, containing 414 pairs with human thematic fit ratings, where role-filling nouns were selected to reflect a wide distribution of scores for each verb. Example pairs from both data sets are provided in Table 2.2. For instance, it is more likely for an artist to draw something (draw/artist = 6.6) , than it is for an artist to be drawn (artist/draw = 2.1).

| Event pair | Score |
|---|---|
| **GS199** | |
| river meet sea - river satisfy sea | 1.84 |
| user write software - user publish software | 3.92 |
| people run company - people operate company | 6.53 |
| **KS108** | |
| author write book - delegate buy land | 1.13 |
| panel discuss issue - project present problem | 3.73 |
| medication achieve result - drug produce effect | 6.16 |

Table 2.3 Example event pairs and their averaged human similarity scores from GS199 and KS108.

**Event Similarity** While the selectional preference task looks at the SV and VO word pair combinations in isolation, *event similarity* evaluates the plausibility of the whole three-word SVO structure (called *event* in the respective literature) (Grefenstette and Sadrzadeh, 2011a; Weber et al., 2018), as well as correlates the semantic similarity of two SVO structures to

[6] Using an example from Sayeed et al. (2016), the human participants were asked "how common is it for a {snake, monster, baby, cat} to frighten someone/something" (agent role) as opposed to "how common is it for a {snake, monster, baby, cat} to be frightened by someone/something" (patient role).

human-elicited similarity judgments. Robust and flexible event representations are important to many core areas in language understanding such as script learning, narrative generation, and discourse understanding (Chambers and Jurafsky, 2009; Pichotta and Mooney, 2016; Modi, 2016; Weber et al., 2018).

Two benchmarking data sets are typically used for event similarity evaluations: **GS199** (Grefenstette and Sadrzadeh, 2011a) and **KS108** (Kartsaklis and Sadrzadeh, 2014). GS199 contains 199 pairs of $SVO$ triplets/events. In the GS199 data set only the $V$ component is varied, while $S$ and $O$ are fixed in the pair: this evaluation prevents the model from relying only on simple lexical overlap for similarity computation.[7] KS108 contains 108 event pairs for the same task, but is specifically constructed without any lexical overlap between the events in each pair. Table 2.3 shows examples from both evaluation sets along with their averaged human scores.

We make use of these existing benchmarks in Chapter 6, where we introduce a new function-specific representation learning model.

## 2.2 Models for Word Representations and Relations

### 2.2.1 General Word Representations

Standard word representation models such as skip-gram negative sampling (SGNS) (Mikolov et al., 2013b,a), Glove (Pennington et al., 2014), or FastText (Bojanowski et al., 2017) induce a single word embedding space (Schütze, 1993) capturing broad semantic relatedness (Hill et al., 2015). For instance, SGNS makes use of two vector spaces for this purpose, which are referred to as $A_w$ and $A_c$. SGNS is typically trained using word co-occurences in large text corpora, such that word vectors will lie close together if they tend to have similar surrounding context words. The SGNS objective arranges word vectors ($A_w$) such that their position will predict which context words ($A_c$) they frequently co-occur with. In other words, the dot-product between a word vector (from $A_w$) and a context vector (from $A_c$) will be high if those two words frequently co-occur in the training data.

**Matrix Factorisation and Choice of Context** SGNS has been shown to approximately correspond to factorising a matrix $M = A_w A_c^T$, where elements in $M$ represent the co-occurrence strengths between *words* and their *context* words (Levy and Goldberg, 2014b). Both matrices represent the same vocabulary: therefore, only one of them is needed in prac-

[7] For instance, the phrases *'people run company'* and *'people operate company'* have a high similarity score of 6.53, whereas *'river meet sea'* and *'river satisfy sea'* have been given a low score of 1.84.

tice to represent each word. Typically only $A_w$ is used while $A_c$ is discarded, or the two vector spaces are averaged to produce the final space. For training however, the choice of contexts is crucial to determine the arrangement of resulting word vectors in $A_w$. For instance, Levy and Goldberg (2014a) use dependency-based contexts and show that they produce markedly different embeddings. Schwartz et al. (2015) demonstrate that training on co-occurrences extracted from symmetric patterns (e.g. "X *and* Y") can lead to high performance on semantic similarity datasets. FastText (Bojanowski et al., 2017) enriches word vectors with subword information using a bag of character n-gram vectors. Here, word vectors are replaced by the sum of their subword character n-gram vectors, leading to subword-aware representations.

What these models have in common is that they learn a single space, targeting semantic similarity. Our work in Chapter 6 experiments with using different vocabularies for $A_w$ and $A_c$, while optimising vectors in both matrices to be used in downstream tasks (e.g. learning the SVO structure for selectional preferences and event similarity).

### 2.2.2 Compositional Representations

**Neuroscience.** Theories from cognitive linguistics and neuroscience reveal that common single-space representation models fail to adequately reflect the organisation of semantic concepts in the human brain (i.e., *semantic memory*): there seems to be no single semantic system indifferent to modalities or categories in the brain (Riddoch et al., 1988). Recent fMRI studies strongly support this proposition and suggest that semantic memory is in fact a widely distributed neural network (Davies et al., 2009; Huth et al., 2012; Pascual et al., 2015; Rice et al., 2015; de Heer et al., 2017), where sub-networks might activate selectively or more strongly for a particular function such as modality-specific or category-specific semantics (such as objects/actions, abstract/concrete, animate/inanimate, animals, fruits/vegetables, colours, body parts, countries, flowers, etc.) (Warrington, 1975; Warrington and McCarthy, 1987; McCarthy and Warrington, 1988). This indicates a *function-specific* division of lower-level semantic processing. Single-space distributional word models have been found to partially correlate to these distributed brain activity patterns (Mitchell et al., 2008; Huth et al., 2012, 2016; Anderson et al., 2017), but fail to explain the full spectrum of fine-grained word associations humans are able to make.

**Compositional Distributional Semantics.** Partially motivated by similar observations, prior work frequently employs tensor-based methods for composing separate tensor spaces (Coecke et al., 2010): there, syntactic categories are often represented by tensors of different orders based on assumptions on their relations. One fundamental difference is made between

atomic types (e.g., nouns) versus compositional types (e.g., verbs). Atomic types are seen as standalone: their meaning is independent from other types. On the other hand, verbs are compositional as they rely on their subjects and objects for their exact meaning. Due to this added complexity, the compositional types are often represented with more parameters than the atomic types, e.g., with a matrix instead of a vector. The goal is then to compose constituents into a semantic representation which is independent of the underlying grammatical structure. Therefore, a large body of prior work is concerned with finding appropriate composition functions (Grefenstette and Sadrzadeh, 2011a,b; Kartsaklis et al., 2012; Milajevs et al., 2014) to be applied on top of word representations. Since this approach represents different syntactic structures with tensors of varying dimensions, comparing syntactic constructs is not straightforward. This compositional approach thus struggles with transferring the learned knowledge to downstream tasks.

State-of-the-art compositional models (Tilk et al., 2016; Weber et al., 2018) combine similar tensor-based approaches with neural training, leading to task-specific compositional solutions. While effective for a task at hand, the resulting models rely on a large number of parameters and are not robust: we observe deteriorated performance on other related compositional tasks, as shown in Section 5.10.

**Multivariable (SVO) Structures in NLP.** Modeling SVO-s is important for tasks such as compositional *event similarity* using all three variables, and *thematic fit* modeling based on *SV* and *VO* associations separately. Traditional solutions are typically based on clustering of word co-occurrence counts from a large corpus (Baroni and Lenci, 2010; Greenberg et al., 2015a,b; Sayeed et al., 2016; Emerson and Copestake, 2016). More recent solutions combine neural networks with tensor-based methods. Van de Cruys (2014) present a feedforward neural net trained to score compositions of both two and three groups with a max-margin loss. Grefenstette and Sadrzadeh (2011a,b); Kartsaklis and Sadrzadeh (2014); Milajevs et al. (2014); Edelstein and Reichart (2016) employ tensor compositions on standard single-space word vectors. Hashimoto and Tsuruoka (2016) discern compositional and non-compositional phrase embeddings starting from HPSG-parsed data.

## 2.3 Language Modeling

We now move on to discuss language modelling (LM), a task of slightly different nature than the others presented in this chapter. Language Modeling is a key NLP task, and serves as an important component for applications that require some form of text generation, such as machine translation (Vaswani et al., 2013), speech recognition (Mikolov et al., 2010), dialogue generation (Serban et al., 2016), or summarisation (Filippova et al., 2015).

A language model computes a probability distribution over sequences of tokens, and is typically trained to maximise the likelihood of token input sequences (Chen and Goodman, 1999; Bengio et al., 2003). For the purpose of this dissertation we look at next-word prediction, and therefore adopt token to mean word. The LM objective is expressed as:

$$P(w_1, ...w_n) = \prod_i P(w_i|w_1, ...w_{i-1}) \tag{2.1}$$

$w_i$ is a word token with the index $i$ in the sequence. LM is considered a central task in NLP and language understanding, with applications in speech recognition (Mikolov et al., 2010), text summarisation (Filippova et al., 2015; Rush et al., 2015), and information retrieval (Ponte and Croft, 1998; Zamani and Croft, 2016). The importance of language modeling has been accentuated even more in representation learning recently, where it is used as a novel form of unsupervised pre-training (and an alternative to static word embeddings) for the benefit of a variety of NLP applications (Peters et al., 2018; Howard and Ruder, 2018a).

Language modelling as a task that specifically refers to modeling language in the sequential way given by equation 2.2.[8] Instead of comparing to human annotations to evaluate the quality of a language model, the most typical measure used is **Perplexity (PPL)**. Perplexity evaluates the ability of a trained model to correctly predict the sequence of words in a test corpus. It is the inverse probability of the test set, normalized by the number of words. (Jurafsky and Martin, 2017, Chapter 4.2.1) For a corpus $T = w_1, w_2, ..., w_N$:

$$PPL(T) = P(w_1, w_2, ..., w_N)^{-\frac{1}{N}} \tag{2.2}$$

where $w_1, w_2, ..., w_N$ is the sequence of words given by the corpus. The probability for a word $w_N$ at each time step is calculated as a distribution over all words in the vocabulary. This evaluation metric thereby gives an easy point of comparison if models are trained and evaluated on exactly the same data, with exactly the same vocabulary. Unfortunately in practice this often is not the case, which can lead to a number of comparability issues. In particular: (1) this metric is linking the evaluation to the model itself. The vocabulary needs to be kept the same for backward comparability to previous models. This makes it difficult to provide a fair comparison to more flexible models that can take subword units into account or predict characters. (2) this entirely word-based model and evaluation is suboptimal especially for morphologically-rich languages, where many words are rare since they are simply a

[8] More recent approaches such as Peters et al. (2018); Howard and Ruder (2018a) employ a language modeling objective as a way of pre-training rather than for the LM task itself, and therefore may adapt a more lenient formulation that trains in a bi-directional way.

morphological variant of an otherwise frequent word. This makes language modelling for morphologically-rich languages a challenging problem, both for model design and evaluation.

**Datasets** Language modeling is predominantly tested on English and other Western European languages. Standard English LM benchmarks are the Penn Treebank (PTB) (Marcus et al., 1993) and the 1 Billion Word Benchmark (BWB) (Chelba et al., 2013). Datasets extracted from BBC News (Greene and Cunningham, 2006) and IMDB Movie Reviews (Maas et al., 2011) are also used for LM evaluation in English (Wang and Cho, 2016; Miyamoto and Cho, 2016; Press and Wolf, 2017).

For multilingual LM evaluation, Botha and Blunsom (2014) extract datasets for Czech, French, Spanish, German, and Russian from the 2013 Workshop on Statistical Machine Translation (WMT) data (Bojar et al., 2013). Kim et al. (2016) reuse these datasets and add Arabic. Ling et al. (2015) evaluate on English, Portuguese, Catalan, German and Turkish datasets extracted from Wikipedia. Kawakami et al. (2017) evaluate on 7 European languages using Wikipedia data, including Finnish. To the best of our knowledge, the largest datasets used in previous work are from (Müller et al., 2012; Cotterell et al., 2018) and amount to 21 languages from the Europarl data (Koehn, 2005). Despite the large coverage of languages, these sets are still restricted only to the languages of the European Union. On the other hand, the most typologically diverse dataset thus far was released by Vania and Lopez (2017). It includes 10 languages representing some morphological systems.

This short survey of existing datasets demonstrates a clear tendency towards extending LM evaluation to other languages, abandoning English-centric assumptions, and focusing on language-agnostic LM architectures. However, a comprehensive evaluation set that systematically covers a wide and balanced spectrum of typologically diverse languages is still missing. We introduce a novel dataset aimed at bridging this gap, as well as extensively discuss modeling challenges especially for morphologically-rich languages in Part III.

## 2.4 Conclusions

In this chapter we introduced a number of key topics and tasks which we will build on in the following chapters. It is important to reiterate that most of these tasks (with the exception of language modeling) are entirely independent from the actual model used to solve them. These tasks can be seen as giving *guidelines* for model development based on human annotations. Further it is important to note that while some of these tasks may overlap, in many instances *word pairs might score entirely different depending on the task*: For example, the word pair *cat - dog* should receive a high score for semantic similarity since both are common pet

animals, but a moderate or low score for entailment since there is no type-of relationship between them; they are in the same level of the entailment hierarchy. Equally, *cat - eat* should score high for subject-verb selectional preference, but low for both semantic similarity and entailment.

Ideally, one model should be able to jointly capture all of these semantic relations. As we show in Part II, to date different models will work best for different relations. In Chapter 6 we develop a model to jointly address selectional preference and event similarly, which has traditionally been treated with task-specific models. Part III gives perspective on requirements for modeling further languages. Future work may join these approaches together and look at modeling semantic relations for a broader set of languages.

# Part II

# Semantic Relations

# Motivation

Words can relate to each other in a myriad of ways. Most commonly, word representations are trained based on the distributional hypothesis, resulting in general representations where words with similar meaning are close to each other in the induced vector space (Harris, 1954; Firth, 1968). However, this type of representation has been found to conflate several useful semantic relations (Mrkšić et al., 2017). Equally, intrinsic evaluation sets for general semantic representations can be annotated according to different standards, and thereby might conflate multiple useful relations. In particular, often a distinction between *relatedness* and *similarity* is not made. Many data sets focus on evaluating *relatedness* for nouns in particular, conflated with true similarity, and a variety of other semantic relations (Hill et al., 2015).

While noun relatedness is, without a doubt, a one major category to cover, a variety of word types and semantic relations remain without thorough intrinsic evaluation sets. Intrinsic evaluation sets are intended to give guidance for developing models. A consequence of narrow coverage thereby is that models are being built to perform well for noun relatedness only. However, this has been found to not correlate well to downstream task performance (Chiu et al., 2016). Simultaneously, model performance on other word types and semantic relations, as well as its impact on downstream task performance remains unclear. This suggests there is a strong need for a broader scope of evaluation resources, covering a wider range of linguistic phenomena.

In this part we primarily look at semantic similarity for verbs. Semantic similarity has been found to correlate well to performance on downstream tasks (Chiu et al., 2016), and verbs are usually investigated less in favour of nouns (Section 3.2.1), despite their semantics being critical for language understanding. As a second semantic relation we will also look at lexical entailment (Chapter 4).

The work on verb semantic similarity has been published as:

**Daniela Gerz**, Ivan Vulić, Felix Hill, Roi Reichart and Anna Korhonen. "SimVerb-3500: A large-scale evaluation set of verb similarity." *EMNLP 2016*.

Additionally this part contains work done in collaboration. This work has been published in the following papers:

- Ivan Vulić, **Daniela Gerz**, Douwe Kiela, Felix Hill, and Anna Korhonen. "Hyperlex: A large-scale evaluation of graded lexical entailment." *Computational Linguistics*, 43(4).
- Marek Rei, **Daniela Gerz**, and Ivan Vulić. "Scoring Lexical Entailment with a Supervised Directional Similarity Network." *ACL 2018*.

For Hyperlex, I mainly provided help with computation, benchmarking against previous models and baselines, and partially with the design of the evaluation resource. Chapter 4.6, in particular the SDSN model, is a contribution of Marek Rei.

# Chapter 3

# Evaluating Verb Similarity: SimVerb-3500

We introduce SimVerb-3500, an evaluation resource that provides human ratings for the similarity of 3,500 verb pairs. SimVerb-3500 covers all normed verb types from the USF free-association database, providing at least three examples for every VerbNet class. This broad coverage facilitates detailed analyses of how syntactic and semantic phenomena together influence human understanding of verb meaning. Further, with significantly larger development and test sets than existing benchmarks, SimVerb-3500 enables more robust evaluation of representation learning architectures and promotes the development of methods tailored to verbs. We hope that SimVerb-3500 will enable a richer understanding of the diversity and complexity of verb semantics and guide the development of systems that can effectively represent and interpret this meaning.

## 3.1 Introduction

Numerous algorithms for acquiring word representations from text and/or more structured knowledge bases have been developed in recent years (Mikolov et al., 2013a; Pennington et al., 2014; Faruqui et al., 2015). These representations (or *embeddings*) typically contain powerful features that are applicable to many language applications (Collobert and Weston, 2008; Turian et al., 2010). Nevertheless, the predominant approaches to distributed representation learning apply a single learning algorithm and representational form for all words in a vocabulary. This is despite evidence that applying different learning algorithms to word types

such as nouns, adjectives and verbs can significantly increase the ultimate usefulness of representations (Schwartz et al., 2015).

One factor behind the lack of more nuanced word representation learning methods is the scarcity of satisfactory ways to evaluate or analyse representations of particular word types. Resources such as MEN (Bruni et al., 2014), Rare Words (Luong et al., 2013) and SimLex-999 (Hill et al., 2015) focus either on words from a single class or small samples of different word types, with automatic approaches already reaching or surpassing the inter-annotator agreement ceiling. Consequently, for word classes such as *verbs*, whose semantics are critical for language understanding, it is practically impossible to achieve statistically robust analyses and comparisons between different representation learning architectures.

To overcome this barrier to verb semantics research, we introduce *SimVerb-3500* – an extensive intrinsic evaluation resource that is unprecedented in both size and coverage. SimVerb-3500 includes 827 verb types from the University of South Florida Free Association Norms (USF) (Nelson et al., 2004), and at least 3 member verbs from each of the 101 top-level VerbNet classes (Kipper et al., 2008). This coverage enables researchers to better understand the complex diversity of syntactic-semantic verb behaviours, and provides direct links to other established semantic resources such as WordNet (Miller, 1995) and PropBank (Palmer et al., 2005). Moreover, the large standardised development and test sets in SimVerb-3500 allow for principled tuning of hyperparameters, a critical aspect of achieving strong performance with the latest representation learning architectures.

In Chapter 2, we discuss previous evaluation resources targeting verb similarity [9]. Nevertheless, we find that all available datasets of this kind are insufficient for judging verb similarity due to their small size or narrow coverage of verbs. We present the new SimVerb-3500 data set along with our design choices and the pair selection process in Section 3.2, while the annotation process is detailed in Section 3.3. In Section 3.4 we report the performance of a diverse range of popular representation learning architectures, together with benchmark performance on existing evaluation sets. In Section 3.5, we show how SimVerb-3500 enables a variety of new linguistic analyses, which were previously impossible due to the lack of coverage and scale in existing resources.

In this paper we provide a remedy for this problem by presenting a more comprehensive and representative verb pair evaluation resource.

[9] In some existing evaluation sets pairs are scored for relatedness which has some overlap with similarity. SimVerb-3500 focuses on similarity as this is a more focused semantic relation that seems to yield a higher agreement between human annotators. For a broader discussion see (Hill et al., 2015).

## 3.2 The SimVerb-3500 Data Set

In this section, we discuss the design principles behind SimVerb-3500. We first demonstrate that a new evaluation resource for verb similarity is a necessity. We then describe how the final verb pairs were selected with the goal to be representative, that is, to guarantee a wide coverage of two standard semantic resources: USF and VerbNet.

### 3.2.1 Design Motivation

Hill et al. (2015) argue that comprehensive high-quality evaluation resources have to satisfy the following three criteria: *(C1) Representative* (the resource covers the full range of concepts occurring in natural language); *(C2) Clearly defined* (it clearly defines the annotated relation, e.g., similarity); (C3) *Consistent and reliable* (untrained native speakers must be able to quantify the target relation consistently relying on simple instructions).

Building on the same annotation guidelines as Simlex-999 that explicitly targets similarity, we ensure that criteria C2 and C3 are satisfied. However, even SimLex, as the most extensive evaluation resource for verb similarity available at present, is still of limited size, spanning only 222 verb pairs and 170 distinct verb lemmas in total (we refer to this subset as the dataset SL-222). Given that 39 out of the 101 top-level VerbNet classes are not represented at all in SimLex, while 20 classes have only one member verb,[10] one may conclude that the criterion C1 is not at all satisfied with current resources.

There is another fundamental limitation of all current verb similarity evaluation resources: automatic approaches have reached or surpassed the inter-annotator agreement ceiling. For instance, while the average pairwise correlation between annotators on SL-222 is Spearman's $\rho$ correlation of 0.717, the best performing automatic system reaches $\rho = 0.727$ (Mrkšić et al., 2016). SimVerb-3500 does not inherit this anomaly (see Table 3.2) and demonstrates that there still exists an evident gap between the human and system performance.

In order to satisfy C1-C3, the new SimVerb-3500 evaluation set contains similarity ratings for 3,500 *verb pairs*, containing 827 verb types in total and 3 member verbs for each top-level VerbNet class. The rating scale goes from 0 (not similar at all) to 10 (synonymous). We employed the SimLex-999 annotation guidelines. In particular, we instructed annotators to give low ratings to antonyms, and to distinguish between similarity and relatedness. Pairs that are related but not similar (e.g., *to snore / to snooze*, *to walk / to crawl*) thus have a fairly low rating. Several example pairs are provided in Table 3.1.

[10] Note that verbs in VerbNet are soft clustered, and one verb type may be associated with more than one class. When computing coverage, we assume that such verbs attribute to counts of all their associated classes.

| Pair | Rating |
|---|---|
| to reply / to respond | 9.79 |
| to snooze / to nap | 8.80 |
| to cook / to bake | 7.80 |
| to participate / to join | 5.64 |
| to snore / to snooze | 4.15 |
| to walk / to crawl | 2.32 |
| to stay / to leave | 0.17 |
| to snooze / to happen | 0.00 |

Table 3.1 Example verb pairs from SimVerb-3500.

### 3.2.2 Choice of Verb Pairs and Coverage

To ensure a wide coverage of a variety of syntactico-semantic phenomena (C1), the choice of verb pairs is steered by two standard semantic resources available online: (1) the USF norms data set[11] (Nelson et al., 2004), and (2) the VerbNet verb lexicon[12] (Kipper et al., 2004, 2008).

The **USF** norms data set (further USF) is the largest database of free association collected for English. It was generated by presenting human subjects with one of 5,000 cue concepts and asking them to write the first word coming to mind that is associated with that concept. Each cue concept $c$ was normed in this way by over 10 participants, resulting in a set of associates $a$ for each cue, for a total of over 72,000 $(c,a)$ pairs. For each such pair, the proportion of participants who produced associate $a$ when presented with cue $c$ can be used as a proxy for the strength of association between the two words.

The norming process guarantees that two words in a pair have a degree of semantic association which correlates well with semantic relatedness and similarity. Sampling from the USF set ensures that both related but non-similar pairs (e.g., *to run / to sweat*) as well as similar pairs (e.g., *to reply / to respond*) are represented in the final list of pairs. Further, the rich annotations of the output USF data (e.g., concreteness scores, association strength) can be directly combined with the SimVerb-3500 similarity scores to yield additional analyses and insight.

**VerbNet** (**VN**) is the largest online verb lexicon currently available for English. It is hierarchical, domain-independent, and broad-coverage. VN is organised into verb classes extending the classes from Levin (1993) through further refinement to achieve syntactic and semantic coherence among class members. According to the official VerbNet guidelines,[13]

[11] http://w3.usf.edu/FreeAssociation/

[12] http://verbs.colorado.edu/verb-index/

[13] http://verbs.colorado.edu/verb-index/VerbNet_Guidelines.pdf

"Verb Classes are numbered according to shared semantics and syntax, and classes which share a top-level number (9-109) have corresponding semantic relationships." For instance, all verbs from the top-level Class 9 are labelled "Verbs of Putting", all verbs from Class 30 are labelled "Verbs of Perception", while Class 39 contains "Verbs of Ingesting".

Among others, three basic types of information are covered in VN: (1) verb subcategorization frames (SCFs), which describe the syntactic realization of the predicate-argument structure (e.g. *the window broke*), (2) selectional preferences (SPs), which capture the semantic preferences verbs have for their arguments (e.g. *a breakable physical object* broke) and (3) lexical-semantic verb classes (VCs) which provide a shared level of abstraction for verbs similar in their (morpho-)syntactic and semantic properties (e.g. *BREAK verbs*, sharing the VN class 45.1, and the top-level VN class 45).[14] The basic overview of the VerbNet structure already suggests that measuring verb similarity is far from trivial as it revolves around a complex interplay between various semantic and syntactic properties.

The wide coverage of VN in SimVerb-3500 assures the wide coverage of distinct verb groups/classes and their related linguistic phenomena. Finally, VerbNet enables further connections of SimVerb-3500 to other important lexical resources such as FrameNet (Baker et al., 1998), WordNet (Miller, 1995), and PropBank (Palmer et al., 2005) through the sets of mappings created by the SemLink project initiative (Loper et al., 2007).[15]

**Sampling Procedure** We next sketch the complete sampling procedure which resulted in the final set of 3500 distinct verb pairs finally annotated in a crowdsourcing study (Section 3.3).

**(Step 1)** We extracted all possible verb pairs from USF based on the associated POS tags available as part of USF annotations. To ensure that semantic association between verbs in a pair is not accidental, we then discarded all such USF pairs that had been associated by two or fewer participants in USF.

**(Step 2)** We then manually cleaned and simplified the list of pairs by removing all pairs with multi-word verbs (e.g., *quit / give up*), all pairs that contained the non-infinitive form of a verb (e.g., *accomplished / finished*, *hidden / find*), removing all pairs containing at least one auxiliary verb (e.g., *must / to see, must / to be*). The first two steps resulted in 3,072 USF-based verb pairs.

**(Step 3)** After this stage, we noticed that several top-level VN classes are not part of the extracted set. For instance, 5 VN classes did not have any member verbs included, 22 VN classes had only 1 verb, and 6 VN classes had 2 verbs included in the current set.

[14] https://verbs.colorado.edu/verb-index/vn/break-45.1.php

[15] https://verbs.colorado.edu/semlink/

We resolved the VerbNet coverage issue by sampling from such 'under-represented' VN classes directly. Note that this step is not related to USF at all. For each such class we sampled additional verb types until the class was represented by 3 or 4 member verbs (chosen randomly).[16] Following that, we sampled at least 2 verb pairs for each previously 'under-represented' VN class by pairing 2 member verbs from each such class. This procedure resulted in 81 additional pairs, now 3,153 in total.

**(Step 4)** Finally, to complement this set with a sample of entirely unassociated pairs, we followed the SimLex-999 setup. We paired up the verbs from the 3,153 associated pairs at random. From these random parings, we excluded those that coincidentally occurred elsewhere in USF (and therefore had a degree of association). We sampled the remaining 347 pairs from this resulting set of unassociated pairs.

**(Output)** The final SimVerb-3500 data set contains 3,500 verb pairs in total, covering all associated verb pairs from USF, and (almost) all top-level VerbNet classes. All pairs were manually checked post-hoc by the authors plus 2 additional native English speakers to verify that the final data set does not contain unknown or invalid verb types.

**Frequency Statistics** The 3,500 pairs consist of 827 distinct verbs. 29 top-level VN classes are represented by 3 member verbs, while the three most represented classes cover 79, 85, and 93 member verbs. 40 verbs are not members of any VN class.

We performed an initial frequency analysis of SimVerb-3500 relying on the BNC counts available online (Kilgarriff, 1997).[17] After ranking all BNC verbs according to their frequency, we divided the list into quartiles: Q1 (most frequent verbs in BNC) - Q4 (least frequent verbs in BNC). Out of the 827 SimVerb-3500 verb types, 677 are contained in Q1, 122 in Q2, 18 in Q3, 4 in Q4 (*to enroll, to hitchhike, to implode, to whelp*), while 6 verbs are not covered in the BNC list. 2,818 verb pairs contain Q1 verbs, while there are 43 verb pairs with both verbs absent in Q1. Further empirical analyses are provided in Section 3.5.[18]

## 3.3 Word Pair Scoring

We employ the Prolific Academic (PA) crowdsourcing platform,[19] an online marketplace very similar to Amazon Mechanical Turk and to CrowdFlower.

[16]The following three VN classes are exceptions: (1) Class 56, consisting of words that are dominantly tagged as nouns, but can be used as verbs exceptionally (e.g., *holiday, summer, honeymoon*); (2) Class 91, consisting of 2 verbs (*count, matter*); (3) Class 93, consisting of 2 single word verbs (*adopt, assume*).

[17]https://www.kilgarriff.co.uk/bnc-readme.html

[18]Annotations such as VerbNet class membership, relations between WordNet synsets of each verb, and frequency statistics are available as supplementary material.

[19]https://prolific.ac/ (We chose PA for logistic reasons.)

### 3.3.1 Survey Structure

Following the SimLex-999 annotation guidelines, we had each of the 3500 verb pairs rated by at least 10 annotators. To distribute the workload, we divided the 3500 pairs into 70 tranches, with 79 pairs each. Out of the 79 pairs, 50 are unique to one tranche[20], while 20 manually chosen pairs are in all tranches to ensure consistency. The remaining 9 are duplicate pairs displayed to the same participant multiple times to detect inconsistent annotations.

Each annotation set of 79 pairs is given to a different survey participant as to distribute workload. Participants see 7-8 pairs per page. Pairs are rated on a scale of 0-6 by moving a slider [21]. The first page shows 7 pairs, 5 unique ones and 2 from the consistency set. The following pages are structured the same but display one extra pair from the previous page. Participants are explicitly asked to give these duplicate pairs the same rating for quality control. They are able to navigate back and forth to check and adjust their ratings. We use these questions so that we can identify and exclude participants giving several inconsistent answers.

**Checkpoint Questions** The survey contains three control questions in which participants are asked to select the most similar pair out of three choices. For instance, the first checkpoint is: *Which of these pairs of words is the *most* similar? 1. to run / to jog 2. to run / to walk 3. to jog / to sweat.* One checkpoint occurs right after the instructions and the other two later in the survey. The purpose is to check that annotators have understood the guidelines and to have another quality control measure for ensuring that they are paying attention throughout the survey. If just one of the checkpoint questions is answered incorrectly, the survey ends immediately and all scores from the annotator in question are discarded.

**Participants** 843 raters participated in the study, producing over 65,000 ratings. Unlike other crowdsourcing platforms, PA collects and stores detailed demographic information from the participants upfront. This information was used to carefully select the pool of eligible participants. We restricted the pool to native English speakers with a 90% approval rate (maximum rate on PA), of age 18-50, born and currently residing in the US (45% out of 843 raters), UK (53%), or Ireland (2%). 54% of the raters were female and 46% male,

[20]These pairs are randomly assigned to a tranch, without checking for semantic criteria such as VerbNet classes. Annotators might use the rating scale in different ways, and have distinct understandings of particular semantic classes. We report averaged scores in the final dataset, and assume that randomisation will lead to sensible scores on average.

[21]Prior work frequently employs a 0-10 scale. This can be problematic, as it is very fine-grained, and participants might use the scale differently. We therefore opted to use 0-6 for simplicity, but scale all scores linearly to 0-10 in the final dataset for backwards compatibility to prior work.

In this survey we look at **verb similarity**. Two verbs are synonyms if they have very **similar meanings**. Here are some examples of synonym pairs:

- to rent / to lease
- to change / to modify
- to happen / to occur

In practice, word pairs that are not exactly synonymous may still be very similar. Here are some very similar pairs – we could say they are nearly synonyms:

- to make / to manufacture
- to observe / to notice
- to succeed / to achieve

In contrast, although some of the following word pairs are related, they are not very similar. They represent completely **different actions**, or can be considered **opposites** of each other:

- to cook / to eat
- to love / to hate
- to worry / to relax

In this survey, you are asked to compare word pairs and to rate how similar they are on a scale of 0-6 (0 denotes word pairs which are not similar at all, 6 denotes very similar word pairs). Remember, **things that are related are not necessarily similar**.

If you are ever unsure, think back to the examples of synonymous pairs (*to rent / to lease*), and **consider how close the words are (or are not) to being synonymous.**

There is no right answer to these questions. It is perfectly reasonable to use your intuition or gut feeling as a native English speaker, especially when you are asked to rate word pairs that you think are not similar at all.

Figure 3.1 The survey starts with these annotation guidelines as the first page. Immediately afterwards, a checkpoint question is asked to verify understanding of these guidelines.

with the average age of 30. Participants took 8 minutes on average to complete the survey containing 79 questions.[22]

**Annotation Guidelines** Our annotation guidelines are very similar to those used for Simlex-999 (Hill et al., 2015). We start by explaining synonyms, and explicitly instruct annotators to consider *how close words are to being synonymous* for their ratings. Annotators are also explicitly encouraged to give low ratings to antonyms, and when in doubt to rather opt for a lower score. The full guidelines are shown in Figure 3.1.

### 3.3.2 Post-Processing

We excluded ratings of annotators who *(a)* answered one of the checkpoint questions incorrectly (75% of exclusions); *(b)* did not give equal ratings to duplicate pairs; *(c)* showed suspicious rating patterns (e.g., randomly alternating between two ratings or using one single rating throughout). The final acceptance rate was 84%. We then calculated the average of all

[22]High annotation speed is expected with crowd workers, as they are paid by work completed, not time. We therefore place an emphasis on duplicated and checkpoint questions for quality control.

ratings from the accepted raters ( $\geq 10$ ) for each pair. The score was finally scaled linearly from the 0-6 to the 0-10 interval as in (Hill et al., 2015).

## 3.4 Analysis

**Inter-Annotator Agreement** We employ two measures. **IAA-1 (pairwise)** computes the average pairwise Spearman's $\rho$ correlation between any two raters – a common choice in previous data collection in distributional semantics (Padó et al., 2007; Reisinger and Mooney, 2010a; Silberer and Lapata, 2014; Hill et al., 2015).

A complementary measure would smooth individual annotator effects. For this aim, our **IAA-2 (mean)** measure compares the average correlation of a human rater with the average of all the other raters. SimVerb-3500 obtains $\rho = 0.84$ (IAA-1) and $\rho = 0.86$ (IAA-2) (see Table 3.2).[23]

**Vector Space Models** We compare the performance of prominent representation models on SimVerb-3500. We include: (1) unsupervised models that learn from distributional information in text, including the skip-gram negative-sampling model (*SGNS*) with various contexts (*BOW = bag of words; DEPS = dependency contexts*) as in Levy and Goldberg (2014a), the symmetric-pattern based vectors by Schwartz et al. (2015), and count-based PMI-weighted vectors (Baroni et al., 2014); (2) Models that rely on linguistic hand-crafted resources or curated knowledge bases. Here, we use sparse binary vectors built from linguistic resources (*Non-Distributional*, (Faruqui and Dyer, 2015)), and vectors fine-tuned to a paraphrase database (*Paragram*, (Wieting et al., 2015)) further refined using linguistic constraints (*Paragram+CF*, (Mrkšić et al., 2016)). Descriptions of these models are in the supplementary material.

**Comparison to SimLex-999 (SL-222)** 170 pairs from SL-222 also appear in SimVerb-3500. The correlation between the two data sets calculated on the shared pairs is $\rho = 0.91$. This proves, as expected, that the ratings are consistent across the two data sets.

Table 3.3 shows a comparison of models' performance on SimVerb-3500 against SL-222. Since the number of evaluation pairs may influence the results, we ideally want to compare sets of equal size for a fair comparison. Picking one random subset of 222 pairs would bias

[23] Note that although IAA is a common measure, we find differences in the literature regarding post-processing of scores, and the exact IAA calculation (Hill et al., 2015; Pilehvar et al., 2018). Pilehvar et al. (2018) report an IAA of 61.2 for SimVerb-3500. Here we do not adjust individual annotator scores to match the average rating level as in Hill et al. (2015). We also did not separately calculate each tranche, but report both IAAs over all annotations, leading to a potentially optimistic estimate.

| Eval set | IAA-1 | IAA-2 | ALL | TEXT |
|---|---|---|---|---|
| WSIM (203) | 0.67 | 0.65 | 0.79<br>SGNS-BOW | 0.79<br>SGNS-BOW |
| SIMLEX (999) | 0.67 | 0.78 | 0.74<br>Paragram+CF | 0.56<br>SymPat+SGNS |
| SL-222 (222) | 0.72 | - | 0.73<br>Paragram+CF | 0.58<br>SymPat |
| SIMVERB (3500) | 0.84 | 0.86 | 0.63<br>Paragram+CF | 0.36<br>SGNS-DEPS |

Table 3.2 An overview of word similarity evaluation benchmarks. ALL is the current best reported score on each data set across all models (including the models that exploit curated knowledge bases and hand-crafted lexical resources, see supplementary material). TEXT denotes the best reported score for a model that learns solely on the basis of distributional information. All scores are Spearman's $\rho$ correlations.

the results towards the selected pairs, and even using 10-fold cross-validation we found variations up to 0.05 depending on which subsets were used. Therefore, we employ a 2-level 10-fold cross-validation where new random subsets are picked in each iteration of each model.[24] The numbers reported as *CV-222* are averages of these ten 10-fold cross-validation runs. The reported results come very close to the correlation on the full data set for all models.

Most models perform much better on SL-222, especially those employing additional databases or linguistic resources. The performance of the best scoring Paragram+CF model is even on par with the IAA-1 of 0.72. The same model obtains the highest score on SV-3500 ($\rho = 0.628$), with a clear gap to IAA-1 of 0.84. We attribute these differences in performance largely to SimVerb-3500 being a more extensive and diverse resource in terms of verb pairs.

**Development Set** A common problem in scored word pair datasets is the lack of a standard split to development and test sets. Previous works often optimise models on the entire dataset, which leads to overfitting (Faruqui et al., 2016) or use custom splits, e.g., 10-fold cross-validation (Schwartz et al., 2015), which make results incomparable with others. The lack of standard splits stems mostly from small size and poor coverage – issues which we have solved with SimVerb-3500.

Our development set contains 500 pairs, selected to ensure a broad coverage in terms of similarity ranges (i.e., non-similar and highly similar pairs, as well as pairs of medium similarity are represented) and top-level VN classes (each class is represented by at least 1 member verb). The test set includes the remaining 3,000 verb pairs. The performances of representation learning architectures on the dev and test sets are reported in Table 3.3. The

[24]This means 100 runs in total for each number reported. Subsets are randomly selected each time.

| Model | SV-3500 | CV-222 | SL-222 | DEV-500 | TEST-3000 |
|---|---|---|---|---|---|
| SGNS-BOW-PW (d=300) | 0.274 | 0.279 | 0.328 | 0.333 | 0.265 |
| SGNS-DEPS-PW (d=300) | 0.313 | 0.314 | 0.390 | 0.401 | 0.304 |
| SGNS-UDEPS-PW (d=300) | 0.259 | 0.262 | 0.347 | 0.313 | 0.250 |
| SGNS-BOW-8B (d=500) | 0.348 | 0.343 | 0.307 | 0.378 | 0.350 |
| SGNS-DEPS-8B (d=500) | 0.356 | 0.347 | 0.385 | 0.389 | 0.351 |
| SYMPAT-8B (d=500) | 0.328 | 0.336 | 0.544 | 0.276 | 0.347 |
| COUNT-SVD (d=500) | 0.196 | 0.200 | 0.059 | 0.259 | 0.186 |
| NON-DISTRIBUTIONAL | 0.596 | 0.596 | 0.689 | 0.632 | 0.600 |
| PARAGRAM (d=25) | 0.418 | 0.432 | 0.531 | 0.443 | 0.433 |
| PARAGRAM (d=300) | 0.540 | 0.528 | 0.590 | 0.525 | 0.537 |
| PARAGRAM+CF (d=300) | 0.628 | 0.625 | 0.727 | 0.611 | 0.624 |

Table 3.3 Evaluation of state-of-the-art representation learning models on the full SimVerb-3500 set (SV-3500), the Simlex-999 verb subset containing 222 pairs (SL-222), cross-validated subsets of 222 pairs from SV-3500 (CV-222), and the SimVerb-3500 development (DEV-500) and test set (TEST-3000).

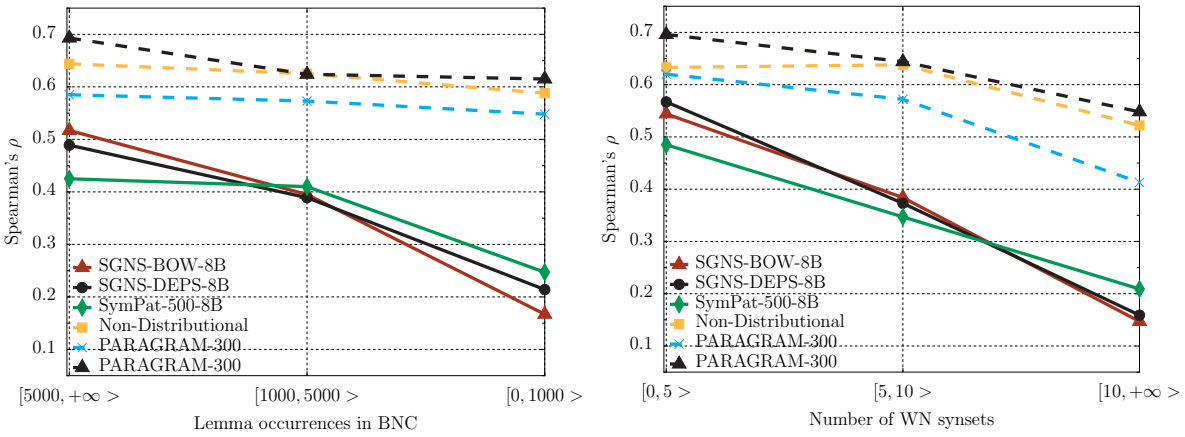
ranking of models is identical on the test and the full SV-3500 set, with slight differences in ranking on the development set.

## 3.5 Evaluating Subsets

The large coverage and scale of SimVerb-3500 enables model evaluation based on selected criteria. In this section, we showcase a few example analyses.

**Frequency** In the first analysis, we select pairs based on their lemma frequency in the BNC corpus and form three groups, with 390-490 pairs in each group (Figure 3.2a). The results from Figure 3.2a suggest that the performance of all models improves as the frequency of the verbs in the pair increases, with much steeper curves for the purely distributional models (e.g., SGNS and SymPat). The non-distributional non data-driven model of Faruqui and Dyer (2015) is only slightly affected by frequency.

**WordNet Synsets** Intuitively, representations for verbs with more diverse usage patterns are more difficult to learn with statistical models. To examine this hypothesis, we resort to WordNet (Miller, 1995), where different semantic usages of words are listed as so-called *synsets*. Figure 3.2b shows a clear downward trend for all models, confirming that polysemous verbs are more difficult for current verb representation models. Nevertheless,




(a) Frequency-based evaluation.

(b) WN synset-based evaluation

Figure 3.2 Subset-based evaluation. **(a)** Subsets are created based on the frequency of verb lemmas in the BNC corpus. Each of the three frequency groups contains 390-490 verb pairs. To be included in each group it is required that both verbs in a pair are contained in the same frequency interval (x axis). **(b)** Subsets are created based on the number of synsets in WordNet (x axis). To be included in each subset it is required that both verbs in a pair have the number of synsets in the same interval.

approaches which use additional information beyond corpus co-occurrence are again more robust. Their performance only drops substantially for verbs with more than 10 synsets, while the performance of other models deteriorates already when tackling verbs with more than 5 synsets.

**VerbNet Classes** Another analysis enabled by SimVerb-3500 is investigating the connection between VerbNet classes and human similarity judgments. We find that verbs in the same top-level VerbNet class are often not assigned high similarity scores. Out of 1378 pairs where verbs share the top-level VerbNet class, 603 have a score lower than 5. Table 3.4 reports scores per VerbNet class. When a verb belongs to multiple classes, we count it for each class (see Footnote 2). We run the analysis on the five largest VN classes, each with more than 100 pairs with paired verbs belonging to the same class.

The results indicate clear differences between classes (e.g., Class 31 vs Class 51), and suggest that further developments in verb representation learning should also focus on constructing specialised representations at the finer-grained level of VN classes.

**Lexical Relations** SimVerb-3500 contains relation annotations (e.g., *antonyms*, *synonyms*, *hyper-/hyponyms*, *no relation*) for all pairs extracted automatically from WordNet. Evaluating

| **Model** | #13 | #31 | #37 | #45 | #51 |
|---|---|---|---|---|---|
| SGNS-BOW-8B | 0.210 | 0.308 | 0.352 | 0.270 | 0.170 |
| SGNS-DEPS-8B | 0.289 | 0.270 | 0.306 | 0.238 | 0.225 |
| SYMPAT-8B (d=500) | 0.171 | 0.320 | 0.143 | 0.195 | 0.113 |
| NON-DISTR | 0.571 | 0.483 | 0.372 | 0.501 | 0.499 |
| PARAGRAM (d=300) | 0.571 | 0.504 | 0.567 | 0.531 | 0.387 |
| PARAGRAM+CF | 0.735 | 0.575 | 0.666 | 0.622 | 0.614 |

Table 3.4 Spearman's $\rho$ correlation between human judgments and model's cosine similarity by VerbNet Class. We chose classes #13 *Verbs of Change of Possession*, #31 *Verbs of Psychological State*, #37 *Verbs of Communication*, #45 *Verbs of Change of State*, and #51 *Verbs of Motion* as examples. All are large classes with more than 100 pairs each, and the frequencies of member verbs are distributed in a similar way.

| **Model** | NR | SYN | HYP |
|---|---|---|---|
| SGNS-BOW-PW (d=300) | 0.096 | 0.288 | 0.292 |
| SGNS-DEPS-PW (d=300) | 0.132 | 0.290 | 0.336 |
| SGNS-BOW-8B (d=500) | 0.292 | 0.273 | 0.338 |
| SGNS-DEPS-8B (d=500) | 0.157 | 0.323 | 0.378 |
| SYMPAT-8B-DENSE (d=300) | 0.225 | 0.182 | 0.265 |
| SYMPAT-8B-DENSE (d=500) | 0.248 | 0.260 | 0.251 |
| NON-DISTRIBUTIONAL | 0.126 | 0.379 | 0.488 |
| PARAGRAM (d=300) | 0.254 | 0.356 | 0.439 |
| PARAGRAM+CF (d=300) | 0.250 | 0.417 | 0.475 |

Table 3.5 Spearman's $\rho$ correlation between human judgments and model's cosine similarity based on pair relation type. Relations are based on WordNet, and included in the dataset. The classes are of different size, 373 pairs with no relation (*NR*), 306 synonym (*SYN*) pairs, and 800 hyper/hyponym (*HYP*) pairs. Frequencies of member verbs are distributed in a similar way.

per-relation subsets, we observe that some models draw their strength from good performance across different relations. Others have low performance on these pairs, but do very well on synonyms and hyper-/hyponyms. Selected results of this analysis are in Table 3.5.[25]

**Human Agreement** Motivated by the varying performance of computational models regarding frequency and ambiguous words with many synsets, we analyse what disagreement effects may be captured in human ratings. We therefore compute the average standard deviation of ratings per subset: $avgstdd(S) = \frac{1}{n}\sum_{p\in S}\sigma(r_p)$, where $S$ is one subset of pairs, $n$ is the number of pairs in this subset, $p$ is one pair, and $r_p$ are all human ratings for this pair.

[25] Evaluation based on Spearman's $\rho$ may be problematic with certain categories, e.g., with antonyms. It evaluates pairs according to their ranking; for antonyms the ranking is arbitrary - every antonym pair should have a very low rating, hence they are not included in Table 3.5. A similar effect occurs with highly ranked synonyms, but to a much lesser degree than with antonyms.

While the standard deviation of ratings is diverse for individual pairs, overall the average standard deviations per subset are almost identical. For both the frequency and the WordNet synset analyses it is around ≈1.3 across all subsets, and with only little difference for the subsets based on VerbNet. The only subsets where we found significant variations is the grouping by relations, where ratings tend to be more similar especially on antonyms (*0.86*) and pairs with no relation (*0.92*), much less similar on synonyms (*1.34*) and all other relations (≈*1.4*). These findings suggest that humans are much less influenced by frequency or polysemy in their understanding of verb semantics compared to computational models.

## 3.6 Conclusions

SimVerb-3500 is a verb similarity resource for analysis and evaluation that is of use to researchers involved in understanding how humans or machines represent the meaning of verbs, and, by extension, scenes, events and full sentences. The size and coverage of syntactico-semantic phenomena in SimVerb-3500 makes it possible to compare the strengths and weaknesses of various representation models via statistically robust analyses on specific word classes.

To demonstrate the utility of SimVerb-3500, we conducted a selection of analyses with existing representation-learning models. One clear conclusion is that distributional models trained on raw text (e.g. SGNS) perform very poorly on low frequency and highly polysemous verbs. This degradation in performance can be partially mitigated by focusing models on more principled distributional contexts, such as those defined by symmetric patterns. More generally, the finding suggests that, in order to model the diverse spectrum of verb semantics, we may require algorithms that are better suited to fast learning from few examples (Lake et al., 2011), and have some flexibility with respect to sense-level distinctions (Reisinger and Mooney, 2010b; Vilnis and McCallum, 2015). In future work we aim to apply such methods to the task of verb acquisition.

Beyond the preliminary conclusions from these initial analyses, the benefit of SimVerb-3500 will become clear as researchers use it to probe the relationship between architectures, algorithms and representation quality for a wide range of verb classes. Better understanding of how to represent the full diversity of verbs should in turn yield improved methods for encoding and interpreting the facts, propositions, relations and events that constitute much of the important information in language.

# Chapter 4

# Graded Lexical Entailment

In the previous chapter we have focused on semantic similarity for an under-represented class of words, verbs. The next step is to move beyond similarity, and towards other fundamental relations between concepts. For achieving human language understanding, similarity is far from being the only relevant semantic relation (Quillian, 1966, 1967; Meyer and Friederici, 2016, inter alia)

Cognitive psychology research has established that typicality and category/class membership are computed in human semantic memory as a gradual relation (Rosch, 1973, 1975; Coleman and Kay, 1981; Medin et al., 1984; Hampton, 2007). However, most NLP research, and existing large-scale inventories of concept category membership (WordNet, DBPedia, etc.) treat category membership and LE as binary (Leacock and Chodorow, 1998; Wu and Palmer, 1994; Weeds et al., 2004; Kotlerman et al., 2010). To address this, we propose a new data set, **HyperLex** (Vulić et al., 2017), containing scores for *graded* lexical entailment between 2,616 concept pairs. Evaluating a range of models on HyperLex reveals a huge gap between human performance and existing modeling approaches.

Furthermore, Rei et al. (2018) introduce the Supervised Directional Similarity Network (SDSN), a novel neural architecture for learning task-specific transformation functions on top of distributional word embeddings. Relying on the limited supervision given by the scores in the HyperLex data set, the architecture is able to generalise and transform a general-purpose distributional vector space to model the relation of lexical entailment. Experiments show excellent performance on scoring graded lexical entailment, raising the state-of-the-art on the HyperLex dataset by approximately 25%.

## 4.1 Introduction

The automatic detection and modelling of lexical entailment has been an area of much focus in natural language processing (Bos and Markert, 2005; Dagan et al., 2006; Baroni et al., 2012; Beltagy et al., 2013, inter alia). However, unlike other semantic relations (such as similarity and relatedness, Chapter 3) that are routinely evaluated with *graded* scores, it has traditionally been treated as an *ungraded* relation.

When communicating, humans intuitively reason about relations between concepts and categories (Quillian, 1967; Collins and Quillian, 1969). For instance, most native speakers of English would agree that *dogs*, *cows*, or *cats* are *animals*, and that *tables* or *pencils* are not. In a conversation about a *cat*, humans are able to quickly perform inference and understand that we are talking about an *animal*, even when this fact is not explicitly mentioned. Nevertheless, for less prototypical lexical concepts such as *dinosaur*, *human being* or *amoeba*, the classification might depend on the perspective or scientific knowledge of the speaker.

The Princeton WordNet lexical database (Miller, 1995; Fellbaum, 1998) is perhaps the best known attempt to formally represent these fundamental relations between concepts. In particular, WordNet has the so-called TYPE-OF or **hyponymy–hypernymy** relation that exists between category concepts such as *animal* and their constituent members such as *cat* or *dog*. However, in WordNet, all semantic relations are represented in a binary way (i.e., concept $X$ entails $Y$) rather than gradual (e.g., $X$ entails $Y$ to a certain degree). This binary treatment is a simplification, as fundamentally TYPE-OF is a graded relation (Rosch, 1973, 1975; Coleman and Kay, 1981; Medin et al., 1984; Lakoff, 1990; Hampton, 2007). It makes appropriate categorisation for less prototypical examples such as *dinosaur* or *amoeba* tricky. Additionally, using an ungraded annotation standard is not ideal for modern vector-based models, that by default allow graded scoring through distances in vector space. For other semantic relations such as semantic similarity (SimLex-999 (Hill et al., 2015) or SimVerb-3500 (Gerz et al., 2016) introduced in Chapter 3), graded scores are the default: These data sets contain averaged ratings produced by multiple human annotators, and thereby present a graded score between two words.

Here we introduce a novel resource, HyperLex, that can be used for the intrinsic evaluation of the ability of vector space models to capture the lexical entailment relation between concepts. Encouraged by high inter annotator agreement scores and evidently large gaps between the human and system performance, we believe that HyperLex will guide the development of a new generation of representation-learning architectures that induce hypernymy/LE-specialized word representations, as opposed to nowadays ubiquitous word representations targeting exclusively semantic similarity and/or relatedness.

## 4.2 Graded Lexical Entailment

**(Proto)typicality, Vagueness, and Graded LE** The graded lexical entailment relation as described by the intuitive question "to what degree is X a type of Y?" encompasses two distinct phenomena described in cognitive science research (cf. Hampton (2007)). First, it can be seen as a measure of **typicality** in graded cognitive categorization (Rosch, 1973, 1975; Medin et al., 1984; Lakoff, 1990), where some instances of a category are more central than others. It measures to what degree some class instance X is a prototypical example of class / concept Y. For instance, when humans are asked to give an example instance of the concept *sport*, it turns out that *football* and *basketball* are more frequently cited than *wrestling*, *chess*, *softball* or *raquetball*. Second, the graded lexical entailment relation also arises when one asks about the applicability of concepts to objects: The boundaries between a category and its instances are much more often fuzzy and vague than unambiguous and clear-cut (Kamp and Partee, 1995). In other words, the **graded membership** (often termed **vagueness**) measures the graded applicability of a concept to different instances. For instance, it is not clear to what extent different objects in our surroundings (e.g., *tables*, *pavements*, *washing machines*, *stairs*, *benches*) could be considered members of the category *chair* despite the fact that such objects can be used as "objects on which one can sit."

In short, graded membership of vagueness quantifies "whether or not and to what degree an instance falls within a conceptual category", whereas typicality reflects "how representative an exemplar is of a conceptual category" (Hampton, 2007). In our crowdsourcing study with non-expert workers, we have deliberately avoided any explicit differentiation between the two phenomena captured by the same intuitive "to-what-degree" question, reducing the complexity of the study design and allowing for free variance in collected data in terms of their quantity and representative concept pairs.

**Definition** The classical definition of *ungraded lexical entailment* is as follows: Given a concept word pair $(X, Y)$, $Y$ is a hypernym of $X$ if and only if $X$ is a type of $Y$, or equivalently every $X$ is a $Y$.[26] On the other hand, *graded lexical entailment* defines the strength of the lexical entailment relation between the two concepts. Given the concept pair $(X, Y)$ and the entailment strength $s$, the triplet $(X, Y, s)$ defines to what degree $Y$ is a hypernym of $X$ (i.e., *to what degree X is a type of Y*), where the degree is quantified by $s$, e.g., to what degree *snake* is a TYPE-OF *animal*. Formally, the graded entailment function $f_{graded}$ defines the following mapping:

[26] Other variants of the same definition replace TYPE-OF with KIND-OF or INSTANCE-OF.

$$f_{graded} : (X, Y) \rightarrow \mathbb{R}_0^+ \tag{4.1}$$

where $f_{graded}$ outputs the strength of the lexical entailment relation $s \in \mathbb{R}_0^+$.

By adopting the graded LE paradigm, HyperLex thus measures the degree of lexical entailment between words $X$ and $Y$ constituting the order-sensitive pair $(X, Y)$. From another perspective, it measures the typicality and graded membership of the instance $X$ for the class/category $Y$. By imposing a threshold $thr$ on $s$, all graded relations may be easily converted to discrete ungraded decisions.

## 4.3 HyperLex

The HyperLex evaluation set contains **noun pairs** (2,163 pairs) and **verb pairs** (453 pairs) annotated for the strength of the lexical entailment relation between the words in each pair. Since the LE relation is asymmetric and the score always quantifies to what degree $X$ is a type of $Y$, pairs $(X, Y)$ and $(Y, X)$ are considered distinct pairs. Each concept pair is rated by at least 10 human annotators. The rating scale goes from 0 (no TYPE-OF relationship at all) to 10 (perfect TYPE-OF relationship). Several examples from HyperLex are provided in Table 4.1.

| Pair | HyperLex LE Rating |
|---|---|
| chemistry / science | 10.0 |
| motorcycle / vehicle | 9.85 |
| pistol / weapon | 9.62 |
| to ponder / to think | 9.40 |
| to scribble / to write | 8.18 |
| gate / door | 6.53 |
| thesis / statement | 6.17 |
| to overwhelm / to defeat | 4.75 |
| shore / beach | 3.33 |
| vehicle / motorcycle | 1.09 |
| enemy / crocodile | 0.33 |
| ear / head | 0.00 |

Table 4.1 Example word pairs from HyperLex. The order of words in each pair is fixed, e.g., the pair *chemistry / science* should be read as *"Is CHEMISTRY a type of SCIENCE?"*

In its 2,616 word pairs, HyperLex contains 1,843 distinct noun types and 392 distinct verb types. In comparison, SimLex-999 as the standard crowdsourced evaluation benchmark for representation learning architectures focused on the synonymy relation contains 751 distinct

nouns and 170 verbs in its 999 word pairs. In another comparison, the LE benchmark BLESS (Baroni and Lenci, 2011) contains relations where one of the words in each pair comes from the set of 200 distinct concrete noun types.

### 4.3.1 Choice of Concepts

To ensure a wide coverage of a variety semantic phenomena (C1, Section 3.2.1), the choice of candidate pairs is steered by two standard semantic resources available online: (1) the USF norms data set[27] (Nelson et al., 2004) introduced in Section 3.2.2, and (2) WordNet[28] (Miller, 1995).

The norming process in USF guarantees that two words in a pair have a degree of semantic association which correlates well with semantic relatedness reflected in different lexical relations between words in the pairs. Inspecting the pairs manually revealed a good range of semantic relationship values represented, e.g., there were examples of ungraded LE pairs (*car / vehicle*, *biology / science*), cohyponym pairs (*peach / pear*), synonyms or near-synonyms (*foe / enemy*), meronym–holonym pairs (*heel / boot*), and antonym pairs (*peace / war*). USF also covers different POS categories: nouns (*winter / summer*), verbs (*to elect / to select*), and adjectives (*white / gray*), at the same time spanning word pairs at different levels of concreteness (*panther / cat* vs *wave / motion* vs *hobby / interest*). The rich annotations of the USF data (e.g., concreteness scores, association strength) can be combined with graded LE scores to yield additional analyses and insight.

WordNet was used to automatically assign a fine-grained lexical relation to each pair in the pool of candidates: this guided the sampling process to ensure a wide coverage of word pairs standing in different lexical relations (Shwartz et al., 2016).

**Lexical Relations** To guarantee the coverage of a wide range of semantic phenomena, we have conditioned the cohort/pool used for sampling on the lexical relation between the words in each pair. As mentioned above, the information was extracted from WordNet.[29] We consider the following lexical relations in HyperLex:

1. `hyp-N`: $(X, Y)$ pairs where $X$ is a hyponym of $Y$ according to WordNet. $N$ is the path length between the two concepts in the WordNet hierarchy, e.g., the pair *cathedral / building* is assigned the `hyp-3` relation. Due to unavailability of a sufficient number of pairs for longer paths, we have grouped all pairs with the path length $\geq 4$ into a

[27] http://w3.usf.edu/FreeAssociation/

[28] https://wordnet.princeton.edu/

[29] Lexical relations were used for sampling only. Word pairs are treated equally for scoring regardless of the lexical relation between them.

single class hyp$\geq$4. It was shown that pairs that are separated by fewer levels in the WordNet hierarchy are both more strongly associated and rated as more similar (Hill et al., 2015). This fine-grained division over LE levels enables analyses based on the semantic distance in a concept hierarchy.

2. rhyp-N: The same as hyp-N, now with the order reversed: $X$ is now a hypernym of $Y$. Such pairs were included to investigate the inherent asymmetry of the TYPE-OF relation and how human subjects perceive it.

3. cohyp: $X$ and $Y$ are two instances of the same category, that is, they share a hypernym (e.g., *dog* and *elephant* are instances of the category *animal*). For simplicity, we retain only $(X,Y)$ pairs that share a direct hypernym.

4. mero: It denotes the PART–WHOLE relation, where $X$ always refer to the meronym (i.e., PART), and $Y$ to the holonym (i.e., WHOLE): *finger / hand*, *letter / alphabet*. By its definition, this relation is observed only between nominal concepts.

5. syn: $X$ and $Y$ are synonyms and near-synonyms, e.g., *movement / motion*, *attorney / lawyer*. In case of polysemous concepts, at least one sense has to be synonymous with a meaning of the other concept, e.g., *author / writer*.

6. ant: $X$ and $Y$ are antonyms, e.g., *beginning / end*, *to unite / to divide*.

7. no-rel: $X$ and $Y$ do not stand in any lexical relation, including the ones not present in HyperLex (e.g., causal relations, space–time relations), and are also not semantically related. This relation specifies that there is no any apparent semantic connection between the two concepts at all, e.g., *chimney / swan*, *nun / softball*.

As this listing illustrates, WordNet arranges words into a rigid taxonomy, where either there exists a relation between words or not. While it is possible to extract some notion of a stronger or more distant relation through the path length (e.g. the number of hierarchy levels N for hyp-N pairs), this binary treatment of relations makes appropriate categorisation tricky for less prototypical examples, and is not ideal especially considering modern vector space approaches which allow for fine-grained scoring (Section 4.1). For HyperLex we collected graded annotations, with the aim to better reflect the gradual nature of the TYPE-OF relation. The following sections present a qualitative analysis of the resulting scores.

| TYPE-OF | **animal** | | **food** | | **sport** | | **person** | | **vehicle** |
|---|---|---|---|---|---|---|---|---|---|
| cat | 10.0 | sandwich | 10.0 | basketball | 10.0 | girl | 9.85 | car | 10.0 |
| monkey | 10.0 | pizza | 10.0 | hockey | 10.0 | customer | 9.08 | limousine | 10.0 |
| cow | 10.0 | rice | 10.0 | volleyball | 10.0 | clerk | 8.97 | motorcycle | 9.85 |
| bat | 9.52 | hamburger | 9.75 | soccer | 9.87 | citizen | 8.63 | van | 9.75 |
| mink | 9.17 | mushroom | 9.07 | baseball | 9.75 | nomad | 8.63 | automobile | 9.58 |
| snake | 8.75 | pastry | 8.83 | softball | 9.55 | poet | 7.78 | tractor | 9.37 |
| snail | 8.62 | clam | 8.20 | cricket | 9.37 | guest | 7.22 | truck | 9.23 |
| mongoose | 8.33 | snack | 7.78 | racquetball | 9.03 | mayor | 6.67 | caravan | 8.33 |
| dinosaur | 8.20 | oregano | 5.97 | wrestling | 8.85 | publisher | 6.03 | buggy | 8.20 |
| crab | 7.27 | rabbit | 5.83 | recreation | 2.46 | climber | 5.00 | bicycle | 8.00 |
| plant | 0.13 | dinner | 4.85 | – | – | idol | 4.28 | vessel | 6.38 |

Table 4.2 Graded LE scores for instances of several prominent taxonomical categories/classes represented in HyperLex (i.e., the categories are the word $Y$ in each $(X, Y, s)$ graded LE triplet).

## 4.4 Qualitative Analysis

### 4.4.1 Typicality in Human Judgments

The first straightforward questions to ask are, are some concepts really more (proto)typical of semantically broader higher-level classes? And following from that, can lexical entailment really be treated as a graded relation? Several examples of prominent high-level taxonomic categories along with LE scores are shown in Table 4.2. We might draw several preliminary insights based on the presented lists. There is an evident prototyping effect present in human judgments: concepts such as *cat*, *monkey* or *cow* are more typical instances of the class *animal* than the more peculiar instances such as *mongoose* or *snail* according to HyperLex annotators. Instances of the class *sport* also seem to be sorted accordingly, as higher scores are assigned to arguably more prototypical sports such as *basketball*, *volleyball* or *soccer*, and less prototypical sports such as *racquetball* or *wrestling* are assigned lower scores.

Nonetheless, the majority of `hyp-N` pairs $(X, animal)$ or $(X, sport)$, where $X$ is a hyponym of *animal/sport* according to WN, are indeed assigned reasonably high graded LE scores. It suggests that humans are able to: (1) judge the LE relation consistently and decide that a concept indeed stands in a type-of relation with another concept, and (2) grade the LE relation by assigning more strength to more prototypical class instances. Similar patterns are visible with other class instances from Table 4.2, as well as with other prominent nominal classes (e.g., *bird*, *appliance*, *science*). We also observe the same effect with verbs, e.g., *(drift, move, 8.58), (hustle, move, 7.67), (tow, move, 7.37), (wag, move, 6.80), (unload, move, 6.22)*.

We also analyze if the effects of graded membership are also captured in the ratings, and our preliminary qualitative analysis suggests so. For instance, an interesting example quantifies the graded membership in the class *group*: *(gang, group, 9.25), (legion, group, 7.67), (conference, group, 6.80), (squad, group, 8.33), (caravan, group, 5.00), (grove, group,*

*3.58), (herd, group, 9.23), (fraternity, group, 8.72), (staff, group, 6.28)*. Although we have not explicitly distinguished between typicality and graded membership in our annotation guidelines, with both subsumed under the TYPE-OF formulation of graded lexical entailment, the listed examples suggest that human subjects are able to quantify both in a satisfying manner.

### 4.4.2 Scores By Semantic Relation

Graded LE scores in HyperLex averaged for each WordNet relation are provided in Table 4.3.[30]

The stark differences between average scores of relations make it evident that there are important relation-specific differences. For instance, `ant` pairs are consistently rated low for both nouns (1.57) and verbs (1.25), similar to pairs standing in no relation (0.64 and 1.48 respectively). Graded LE scores for nouns increase with the increase of the LE level (i.e. WN path length) between the concepts. A longer WN path implies a clear difference in semantic generality between nominal concepts which seems to be positively correlated with the degree of the LE relation and ease of human judgment.

| | **All** | **Nouns** | **Verbs** |
|---|---|---|---|
| hyp-1 | 7.86 | 7.99 | 7.49 |
| hyp-2 | 8.10 | 8.31 | 7.08 |
| hyp-3 | 8.16 | 8.39 | 6.55 |
| hyp≥4 | 8.33 | 8.62 | 5.12 |
| cohyp | 3.54 | 3.29 | 4.76 |
| mero | 3.14 | 3.14 | - |
| syn | 6.83 | 6.69 | 7.66 |
| ant | 1.47 | 1.57 | 1.25 |
| no-rel | 0.85 | 0.64 | 1.48 |
| rhyp-1 | 4.75 | 4.17 | 6.45 |
| rhyp-2 | 4.19 | 3.44 | 6.15 |
| rhyp-3 | 3.07 | 2.72 | 4.47 |
| rhyp≥4 | 2.85 | 2.54 | 4.11 |

Table 4.3 Average HyperLex scores across all pairs, and noun and verb pairs representing finer-grained semantic relations extracted from WordNet.

Another factor underlying the observed scores might be the link between HyperLex and the source USF norms. Since USF contains free association norms, one might assume that more prototypical instances are generated more frequently as responses to cue words in the original USF experiments. This, in turn, reflects in their greater presence in HyperLex, especially for concept pairs with longer WN distances.

[30] Note that the LE level is extracted as the shortest direct path between two concept words in the WordNet taxonomy, where $X$-s in each $(X, Y)$ pair always refer to the less general concept (i.e. hyponym).

Further, nominal concepts higher in the WN hierarchy typically refer to semantically very broad but well-defined categories such as *animal*, *food*, *vehicle*, or *appliance* (see again Table 4.2). Semantically more specific instances of such concepts are easier to judge as *true* hyponyms (using the ungraded LE terminology), which also reflects in higher LE ratings for such instances. However, gradience effects are clearly visible even for pairs with longer WN distances (Tab. 4.2).

The behavior with respect to the LE level is reversed for verbs: the average scores decrease over increasing LE levels. We attribute this effect to a higher level of abstractness and ambiguity present in verb concepts higher in the WN hierarchy stemming from a fundamental cognitive difference: Gentner (2006) showed that children find verbs harder to learn than nouns, and Markman and Wisniewski (1997) present evidence that different cognitive operations are used when comparing two nouns or two verbs. For instance, it is intuitive to assume that human subjects find it easier to grade instances of the class *animal* than instances of verb classes such as *to get*, *to set* or *to think*.

For `syn` pairs, we find a medium-high rating on average, pointing to a possible overlap or correlation with semantic similarity (Chapter 3), which we further analyse in the following section.

### 4.4.3 Lexical Entailment vs Similarity

As HyperLex is based on the USF and WordNet, it has some overlapping pairs to existing resources on semantic similarity, SimVerb (Chapter 3), and SimLex (Hill et al., 2015). An important question especially in the light of high average scores of `syn` pairs in HyperLex thereby is how much these resources correlate and capture the same phenomena. Hill et al. (2015) report that there is a correlation between `hyp-N` word pairs and semantic similarity as judged by human raters. For instance, given the same $[0, 10]$ continuous rating scale in SimLex, the average similarity score for SimLex `hyp-1` pairs is 6.62, it is 6.19 for `hyp-2` pairs, and 5.70 for `hyp-3` and `hyp-4`. In fact, the only group scoring higher than `hyp-N` pairs in SimLex-999 are `syn` pairs with the average score of 7.70. In SimVerb we find a similar pattern in that `syn` pairs are the group with the highest average rating of 6.8. For hyp-1-4 grouped together it is 6.01, for `cohyp` it is 4.44. Similarly, `ant` and `no-rel` pairs are consistently rated low, with average ratings of 0.97 and 3.43 in SimVerb, as well as 1.25 and 1.48 in HyperLex respectively. Table 4.4 compares the ratings in HyperLex and SimVerb on selected examples.

We can see that `ant` pairs such as *succeed / fail* consistently receive low ratings in both data sets, and that pairs standing in a relation such as `hyp-1` and `syn` tend to receiver high ratings. However, in these specific cases we can also see that the precise ratings may differ

| Pair | Relation | HyperLex LE Rating | SimVerb Similarity Rating |
|---|---|---|---|
| talk / communicate | hyp-1 | 9.25 | 7.47 |
| integrate / mix | hyp-2 | 9.17 | 6.81 |
| criticize / remark | hyp-2 | 8.33 | 3.15 |
| describe / explain | hyp-4 | 7.17 | 8.13 |
| shine / glow | syn | 6.52 | 8.63 |
| design / draw | cohyp | 5.63 | 5.81 |
| understand / accept | hyp-2 | 4.08 | 7.30 |
| succeed / fail | ant | 0.77 | 1.49 |

Table 4.4 HyperLex ratings compared to SimVerb.

significantly. For example, *shine / glow*, a `syn` pair, received a higher rating in SimVerb (8.63) than HyperLex (6.52). On the contrary, *integrate / mix*, a *hyp-2* pair, received a higher rating in HyperLex (9.17) than SimVerb (6.81). Based on these comparisons, we want to test whether HyperLex really captures the fine-grained and subtle notion of graded lexical entailment, or if the HyperLex annotations were largely driven by decisions at the broader level of semantic similarity. Therefore, we also evaluate state-of-the-art word embedding models obtaining peak scores on SimLex-999 in Sections 4.5.2 and 4.5.3.

## 4.5 Quantitative Evaluation

We now evaluate the performance of a range of classic LE modeling approaches as well as vector space models on HyperLex. Due to a wide variety of models and a large space of results, it is not feasible to present all results, or provide detailed analyses across all potential dimensions of comparison. Equally, for space reasons we omit a description of the referenced models. Please refer to our paper (Vulić et al., 2017) for details on all setups.

### 4.5.1 Experiment I: Ungraded LE Approaches

First, we evaluate a series of state-of-the-art traditional LE modeling approaches in the graded LE task on the entire HyperLex evaluation set. A summary of the results is provided in Table 4.5. Comparing model scores with the inter-annotator agreements suggests that the graded LE task, although well-defined and understandable by average native speakers, poses a challenge for current ungraded LE models. The absolute difference in scores between human and system performance indicates that there is vast room for improvement in future work. The gap also illustrates the increased difficulty of the graded LE task compared to

previous ungraded LE evaluations. For instance, the best unsupervised LE directionality and detection models from Table 4.5 reach up over 70% and up to 90% in precision scores (Santus et al., 2014; Kiela et al., 2015b, inter alia) on BLESS and other ungraded LE data sets.

| Model | **Setup 1** | **Setup 2** |
|---|---|---|
| FR ($\alpha = 0.02, \theta = 0.25$) | 0.279 | 0.240 |
| FR ($\alpha = 0, \theta = 0$) | 0.268 | 0.265 |
| $DEM_1$ | 0.162 | 0.162 |
| $DEM_2$ | 0.171 | 0.180 |
| $DEM_3$ | 0.150 | 0.150 |
| $DEM_4$ | 0.153 | 0.153 |
| SLQS–BASIC | 0.225 | 0.221 |
| SLQS–SIM | 0.228 | 0.226 |
| WN–BASIC | 0.207 | 0.207 |
| WN–LCH | 0.214 | 0.214 |
| WN–WUP | 0.234 | 0.234 |
| VIS–ID ($\alpha = 0.02, \theta = 0$) | 0.203 | 0.203 |
| VIS–CENT ($\alpha = 0.02, \theta = 0$) | 0.209 | 0.209 |
| IAA—1 | 0.854 | 0.854 |
| IAA—2 | 0.864 | 0.864 |

Table 4.5 Results in the graded LE task over all HyperLex concept pairs obtained by the sets of most prominent LE models available in the literature. SETUP 1 and SETUP 2 refer to different training setups for DEMs and SLQS. All results are Spearman's $\rho$ correlation scores. IAA $\rho$ scores are provided to quantify the upper bound for the graded LE task.

Previous work on ungraded LE evaluation also detected that frequency is a surprisingly competitive baseline in LE detection/directionality experiments (Herbelot and Ganesalingam, 2013; Weeds et al., 2014; Kiela et al., 2015b). This finding stems from an assumption that the informativeness of a concept decreases and generality increases as frequency of the concept increases (Resnik, 1995). Although the assumption is a rather big simplification (Herbelot and Ganesalingam, 2013), the results based on simple frequency scores in this work further suggest that a simple concept word frequency ratio (**FR**) model [31] may be used as a very competitive baseline in the graded LE task (Weeds et al., 2004; Santus et al., 2014; Kiela et al., 2015b, inter alia). To our own surprise, the FR model was the strongest model in this first comparison, while directional measures fall short of all other approaches, although prior work suggested that they are tailored to capture the LE relation in particular. As we do not observe any major difference between two setups for **DEM**s (Weeds and Weir, 2003;

[31] FR: $f(X) = freq(X)$, where $freq(X)$ is a word frequency count obtained from a large corpus.

Weeds et al., 2004; Kotlerman et al., 2010) and **SLQS** (Santus et al., 2014), all subsequent experiments use Setup 1. The observed strong correlation between frequency and graded LE supports the intuition that prototypical class instances will be more often cited in text, and therefore simply more frequent.

Even **WN**-based measures (Leacock and Chodorow, 1998; Wu and Palmer, 1994; Pedersen et al., 2004) do not lead to huge improvements over DEMs and fall short of FR. Since WordNet lacks annotations pertinent to the idea of graded LE, such simple WN-based measures cannot quantify the actual LE degree. The inclusion of the basic "semantic relatedness detector" (as controlled by the parameter $\theta$) does not lead to any significant improvements (e.g., as evident from the comparison of SLQS–SIM vs. SLQS–BASIC, or $DEM_2$ vs. $DEM_1$). In summary, the large gap between human and system performances along with the FR superiority over more sophisticated LE approaches from prior work unambiguously calls for the next generation of distributional models tailored for graded lexical entailment in particular.

### 4.5.2 Experiment II: Word Embeddings

In the next experiment, we evaluate a series of state-of-the-art word embedding architectures, covering order embeddings **ORDER-EMB** (Vendrov et al., 2016), and a range of standard embeddings optimized for semantic similarity (Hill et al., 2015): **SGNS-BOW** (Mikolov et al., 2013a,b), **SGNS-DEPS** (Levy and Goldberg, 2014a), **NON-DISTRIBUTIONAL** (Faruqui and Dyer, 2015) **PARAGRAM** (Wieting et al., 2015), and **PARAGRAM+CF** (Mrkšić et al., 2016). A summary of the results is provided in Table 4.6. The scores again reveal the large gap between the system performance and human ability to consistently judge the graded LE relation. The scores on average are similar to or even lower than scores obtained in Exp. I. One trivial reason behind the failure is as follows: word embeddings typically apply the cosine similarity in the Euclidean space to measure the distance between $X$ and $Y$. In practice, this leads to the symmetry: $dist(X,Y) = dist(Y,X)$ for each pair $(X,Y)$, which is an undesired model behavior for graded LE in practice, as human judgements of LE tend to be asymmetric (Vulić et al., 2017). This finding again calls for a new methodology capable of tackling the asymmetry of the graded LE problem in future work.

Dependency-based contexts (SGNS–DEPS) (Levy and Goldberg, 2014a) seem to have a slight edge over ordinary bag-of-words contexts (SGNS–BOW) (Mikolov et al., 2013a,b) which agrees with findings from prior work on ungraded LE (Roller and Erk, 2016; Shwartz et al., 2017). We observe no clear advantage with ORDEREMB (Vendrov et al., 2016), a word embedding models tailored for capturing the hierarchical LE relation naturally in the training objective. We notice slight but encouraging improvements with ORDEREMB when resorting

| Model | **All** | **Nouns** | **Verbs** |
|---|---|---|---|
| FR ($\alpha = 0.02, \theta = 0.25$) | 0.279 | 0.283 | 0.239 |
| FR ($\alpha = 0, \theta = 0$) | 0.268 | 0.283 | 0.091 |
| SGNS–BOW (`win=2`) | 0.167 | 0.148 | 0.289 |
| SGNS–DEPS | 0.205 | 0.182 | 0.352 |
| NON-DISTRIBUTIONAL | 0.158 | 0.115 | 0.543 |
| PARAGRAM | 0.243 | 0.200 | 0.492 |
| PARAGRAM+CF | 0.320 | 0.267 | 0.629 |
| ORDEREMB–COS | 0.156 | 0.162 | 0.005 |
| ORDEREMB–DISTALL | 0.180 | 0.180 | 0.130 |
| ORDEREMB–DISTPOS | 0.191 | 0.195 | 0.120 |
| IAA–1 | 0.854 | 0.854 | 0.855 |
| IAA–2 | 0.864 | 0.864 | 0.862 |

Table 4.6 Results (Spearman's $\rho$ correlation scores) in the graded LE task on HyperLex using a selection of state-of-the-art pre-trained word embedding models. All word embeddings, excluding sparse NON-DISTRIBUTIONAL vectors, are 300-dimensional.

to more sophisticated distance metrics, e.g., moving from the symmetric straightforward COS measure to DISTPOS with ORDEREMB.

### 4.5.3 Model performance on LE vs Similarity

A comparison of average ratings (Section 4.4.3) as well as another look into Table 4.6 indicates an evident link between the LE relation and semantic similarity. Positive correlation scores for all models reveal that pairs with high graded LE scores naturally imply some degree of semantic similarity, e.g., *author / creator*. However, the scores with similarity-specialized models are much lower than the human performance in the graded LE task, which suggests that they cannot capture intricacies of the task accurately. More importantly, there is a dramatic drop in performance when evaluating exactly the same models in the semantic similarity task (i.e., graded synonymy) on SimLex-999 vs. the graded LE task on HyperLex. For instance, two best performing word embedding models on SimLex-999 are PARAGRAM and PARAGRAM+CF reaching Spearman's $\rho$ correlation of 0.685 and 0.742, respectively, with SimLex-999 IAA–1 = 0.673, IAA–2 = 0.778. At the same time, the two models score 0.243 and 0.320 on HyperLex respectively, where the increase in scores for PARAGRAM+CF may be attributed to its explicit control of antonyms through dictionary-based constraints.

A similar decrease in scores is observed with other models in our comparisons, e.g., SGNS–BOW falls from 0.415 on SimLex-999 to 0.167 on HyperLex. To further examine

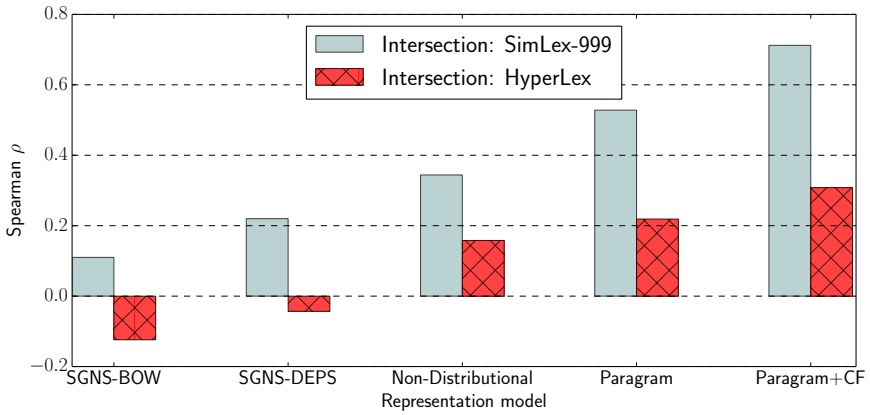



Figure 4.1 Results on the intersection subset of 111 concept pairs annotated both in SimLex-999 (for similarity) and in HyperLex (for graded LE).

this effect, we have performed a simple experiment using only the intersection of the two evaluation sets comprising 111 word pairs in total (91 nouns and 20 verbs) for evaluation. The results of selected embedding models on the 111 pairs are shown in Figure 4.1. It is evident that all state-of-the-art word embedding models are significantly better at capturing semantic similarity.

In summary, the analysis of results with distributed representation models on SimLex-999 and HyperLex suggests that the human understanding of the graded LE relation is not conflated with semantic similarity. Human scores assigned to word pairs in both SimLex-999 and HyperLex reflect truly the nature of the annotated relation: semantic similarity in case of SimLex-999 and graded lexical entailment in case of HyperLex.

## 4.6 From Semantic Representations to Entailment: Specialising Semantic Spaces

In Sections 4.4.3 as well as 4.5.3 we have seen that while semantic similarity and entailment do correlate to a certain degree, the fine-grained notion of graded lexical entailment in fact is different from semantic similarity, as vector spaces with excellent performance on SimLex can perform poorly on HyperLex (Figure 4.1, Table 4.6).

However, recent work in vector/semantic space specialization has shown that it is possible to steer a vector spaces according to explicit linguistic and dictionary knowledge (Yu and Dredze, 2014; Wieting et al., 2015; Faruqui et al., 2015; Astudillo et al., 2015; Liu et al.,

2015; Mrkšić et al., 2016; Vulić et al., 2017, inter alia), and to build vector spaces specialized for capturing different lexical relations, e.g., antonymy (Yih et al., 2012; Ono et al., 2015), or distinguishing between similarity and relatedness (Kiela et al., 2015a). An analogy with (graded) semantic similarity is appropriate here: It was recently demonstrated that vector space models specializing for similarity and scoring high on SimLex-999 and SimVerb-3500 are able to boost performance of statistical systems in language understanding tasks such as *dialogue state tracking* (Mrkšić et al., 2016, 2017; Vulić et al., 2017).

However, purely distributional models coalesce various lexico-semantic relations (e.g., synonymy, antonymy, hypernymy) into a joint distributed representation. To address this, previous work has focused on introducing supervision into *individual* word embeddings, allowing them to better capture the desired lexical properties. For example, Faruqui et al. (2015) and Wieting et al. (2015) proposed methods for using annotated lexical relations to condition the vector space and bring synonymous words closer together. Mrkšić et al. (2016) and Mrkšić et al. (2017) improved the optimisation function and introduced an additional constraint for pushing antonym pairs further apart. While these methods integrate hand-crafted features from external lexical resources with distributional information, they improve only the embeddings of words that have annotated lexical relations in the training resource.

Here, we also look at a novel approach to leveraging external knowledge with general-purpose unsupervised embeddings, focusing on the directional graded lexical entailment task of HyperLex (Vulić et al., 2017), whereas previous work has mostly investigated simpler non-directional semantic similarity tasks. Instead of optimising individual word embeddings, our model published as Rei et al. (2018) uses general-purpose embeddings and optimises a separate neural component to adapt these to the specific task. In particular, the network dynamically produces task-specific embeddings optimised for scoring the asymmetric lexical entailment relation between any two words, regardless of their presence in the training resource. Our results with task-specific embeddings indicate large improvements on the HyperLex dataset. The model also yields improvements on a simpler non-graded entailment detection task.

### 4.6.1 Model

We propose a supervised directional similarity network (SDSN) for mapping vector space trained for semantic similarity to lexical entailment. The network architecture can be seen in Figure 4.2, and the paper (Rei et al., 2018) provides a more fine-grained model description. The system receives a pair of words as input and predicts a score that represents the strength of the given lexical relation. In the graded entailment task, we would like the model to return

a high score for *(biology → science)*, as biology is a type of science, but a low score for *(point → pencil)*.

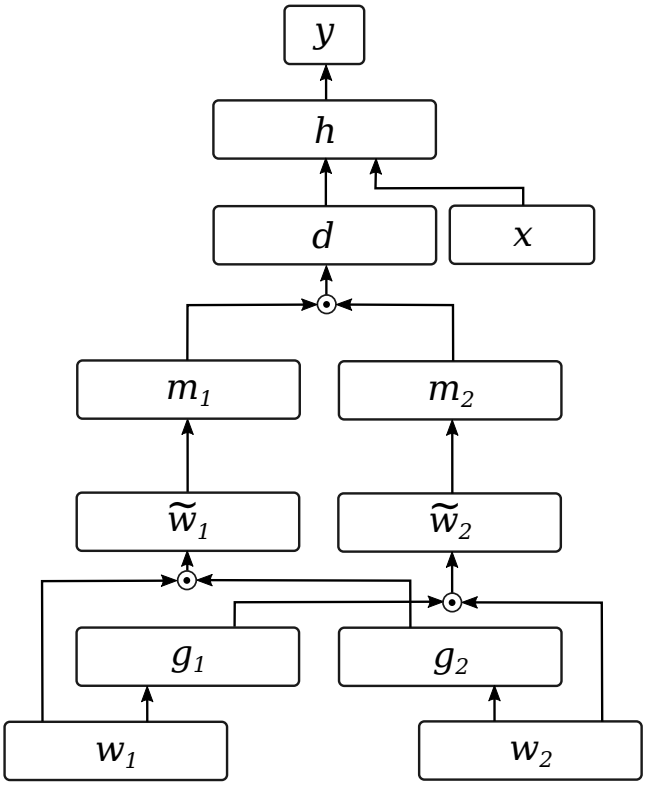



Figure 4.2 Supervised directional similarity network (SDSN) for grading lexical relations.

Both input words are mapped to their corresponding word embeddings $w_1$ and $w_2$. The embeddings come from a standard distributional vector space, pre-trained on a large unannotated corpus, and are not fine-tuned during training. An element-wise gating operation is then applied to each word, conditioned on the other word ($\widetilde{w}_1$, $\widetilde{w}_2$). Each of the word representations is then passed through a non-linear layer with *tanh* activation, mapping the words to a new space that is more suitable for the given task ($m_1$, $m_2$). The vector $d$ then is a dot-product of ($m_1$, $m_2$). For the basic SDSN version of the model, the hidden layer $h$ is a feedforward layer of $d$. For SDSN+SDF we add a 10-dimensional vector $x$, containing corpus-based features. In this version of the model we condition the hidden layer $h$ on the feature vector $x$:

$$h = tanh(W_h d + W_x x + b_h) \tag{4.2}$$

where $x$ is the feature vector of length 10 and $W_x$ is the corresponding weight matrix.

The input embeddings are trained to predict surrounding words on a large unannotated corpus using the skip-gram objective (Mikolov et al., 2013a,b), making the resulting vector space reflect (a broad notion of) semantic relatedness but unsuitable for lexical entailment

(Vulić et al., 2017). The mapping stage allows the network to learn a transformation function from the general skip-gram embeddings to a task-specific space for lexical entailment. In addition, the two weight matrices enable asymmetric reasoning, allowing the network to learn separate mappings for hyponyms and hypernyms. We then use a supervised composition function for combining the two representations and returning a confidence score as output ($y$), similar to Rei et al. (2017). The output $y$ represents the confidence that the two input words are in a lexical entailment relation. The model is optimised by minimising the mean squared distance between the predicted score $y$ and the gold-standard score $\hat{y}$:

$$L = \sum_i (y_i - \hat{y}_i)^2 \tag{4.3}$$

### 4.6.2 Evaluation

**SDSN Training Setup.** As input to the SDSN network we use 300-dimensional dependency-based word embeddings by Levy and Goldberg (2014a). Layers $m_1$ and $m_2$ also have size 300 and layer $h$ has size 100. For regularisation, we apply dropout to the embeddings with $p = 0.5$. The margin $R$ is set to 1 for the supervised pre-training stage. The model is optimised using AdaDelta (Zeiler, 2012) with learning rate 1.0. In order to control for random noise, we run each experiment with 10 different random seeds and average the results. Our code and detailed configuration files are available online.[32]

**SDSN+SDF** Word embeddings are well-suited for capturing distributional similarity, but they have trouble encoding features such as word frequency, or the number of unique contexts the word has appeared in. We construct classical sparse distributional word vectors and use them to extract 5 unique features for every word pair based on the British National Corpus (Leech, 1992), to complement the features extracted from neural embeddings. [33]

**SDSN+SDF+AS** Methods such as retrofitting (RF, Faruqui et al. (2015)), ATTRACT-REPEL (AR, (Mrkšić et al., 2017)) and Poincaré embeddings (Nickel and Kiela, 2017) make use of hand-annotated lexical relations for optimising word representations such that they capture the desired properties (so-called *embedding specialisation*)[34]. We also experiment with incorporating these resources, but instead of adjusting the individual word embeddings, we use them to optimise the shared network weights by using a hinge loss objective. In this setup

[32] http://www.marekrei.com/projects/sdsn

[33] Please refer to the paper (Rei et al., 2018) for the full list of features.

[34] RF and AR fine-tune the vectors given in the resource, the Poincaré approach trains a new space from scratch. We also introduce a variant of AR in Section 5.8.2

[35] we train with a total of 102,586 positive pairs and 42,958 negative pairs extracted from WordNet (Miller, 1995) and the Paraphrase Database (PPDB 2.0) (Pavlick et al., 2015).

**Evaluation Data.** We evaluate graded lexical entailment on the HyperLex dataset (Vulić et al., 2017) which contains 2,616 word pairs in total scored for the asymmetric graded lexical entailment relation. Following a standard practice, we report Spearman's $\rho$ correlation of the model output to the given human-annotated scores. In the *random* split the data is randomly divided into training, validation, and test subsets containing 1831, 130, and 655 word pairs, respectively. In the *lexical split*, proposed by Levy et al. (2015b), there is no lexical overlap between training and test subsets. This prevents the effect of *lexical memorisation*, as supervised models tend to learn an independent property of a single concept in the pair instead of learning a relation between the two concepts. In this setup training, validation, and test sets contain 1133, 85, and 269 word pairs, respectively.[36]

**Results and Analysis** The results on two HyperLex splits are presented in Table 4.7, along with the best configurations reported by Vulić et al. (2017). We refer the interested reader to the original HyperLex paper (Vulić et al., 2017) for a detailed description of the best performing baseline models.

| | Random | | Lexical | |
|---|---|---|---|---|
| | DEV | TEST | DEV | TEST |
| FR | - | 0.299 | - | 0.199 |
| SGNS-DEPS | - | 0.250 | - | 0.253 |
| WN-WuP | - | 0.212 | - | 0.261 |
| SGNS-DEPS (concat+r) | - | 0.539 | - | 0.399 |
| Paragram+CF (cos) | - | 0.346 | - | 0.453 |
| Paragram+CF (mul+r) | - | 0.386 | - | 0.439 |
| SDSN | 0.708 | 0.658 | 0.547 | 0.475 |
| SDSN+SDF | 0.722 | 0.671 | 0.562 | 0.495 |
| SDSN+SDF+AS | **0.757** | **0.692** | **0.577** | **0.544** |

Table 4.7 Graded lexical entailment detection results on the random and lexical splits of the HyperLex dataset. We report Spearman's $\rho$ on both validation and test sets.

[35] Please refer to the paper (Rei et al., 2018) for more details on the setup.

[36] Note that the lexical split discards all cross-set training-test word pairs. Consequently, the number of instances in each subset is lower than with the random split.

The Supervised Directional Similarity Network (**SDSN**) achieves substantially better scores than all other tested systems, despite relying on a much simpler supervision signal: for SDSN and **SDSN+SDF** the designated relation-specific training set and corpus statistics are sufficient. This is an advantage over previous systems which, including the Paragram+CF embeddings, make use of numerous annotations provided by WordNet or similarly rich lexical resources. By adding these extra training instances into our approach (**SDSN+SDF+AS**), we can gain additional performance and push the correlation to 0.692 on the random split and 0.544 on the lexical split of HyperLex, an improvement of approximately 25% to the standard supervised training regime.

For example, the model is able to successfully assign a high score to (*captain*, *officer*) and also identify with high confidence that *wing* is not a type of *airplane*, even though they are semantically related. As an example of incorrect output, the model fails to assign a high score to (*prince*, *royalty*), possibly due to the usage patterns of these words being different in context. In contrast, it assigns an unexpectedly high score to (*kid*, *parent*), likely due to the high distributional similarity of these words.

## 4.7 Conclusion

While the ultimate test of semantic models is their usefulness in downstream applications, the research community is still in need of wide-coverage comprehensive gold standard resources for intrinsic evaluation (Camacho-Collados et al., 2015; Schnabel et al., 2015; Tsvetkov et al., 2015; Hashimoto et al., 2016; Gladkova and Drozd, 2016, inter alia). Such resources can measure the general quality of the representations learned by semantic models, prior to their integration in end-to-end systems. We have presented HyperLex, a large wide-coverage gold standard resource for the evaluation of semantic representations targeting the lexical relation of *graded* lexical entailment (LE) also known as hypernymy-hyponymy or TYPE-OF relation, a relation which is fundamental in construction and understanding of concept hierarchies, that is, semantic taxonomies. Given that the problem of concept category membership is central to many cognitive science problems focused on semantic representation, we believe that HyperLex will also find its use in this domain.

Furthermore we introduce a novel neural architecture for mapping and specialising a vector space based on limited supervision. While prior work has focused only on optimising individual word embeddings available in external resources, our model uses general-purpose embeddings and optimises a separate neural component to adapt these to the specific task, generalising to unseen data. The system achieves new state-of-the-art results on the task

of scoring graded lexical entailment. Future work could apply the model to other lexical relations or extend it to cover multiple relations simultaneously.

Despite the abundance of reported experiments and analyses in this thesis and the corresponding paper (Vulić et al., 2017), we have only scratched the surface in terms of the possible analyses with HyperLex and use of such models as components of broader phrase- and sentence-level textual entailment systems, as well as in other applications. Beyond the preliminary conclusions from these initial analyses, we believe that the benefit of HyperLex will become evident as researchers use it to probe the relationship between architectures, algorithms and representation quality for a wide range of concepts. A better understanding of how to represent the full diversity of concepts (with LE grades attached) in hierarchical semantic networks should in turn yield improved methods for encoding and interpreting the hierarchical semantic knowledge which constitutes much of the important information in language.

# Part III

# Language Modeling for Morphologically-Rich Languages

# Motivation

> Among my most prized possessions are words that I have never spoken.
>
> Orson Rega Card

A key challenge in NLP is developing language-independent methods that can work across a wide variety of languages. This ambition is largely hampered by a lack of adequate resources that span a wide spectrum of typologically-diverse languages.

A large part of the literature on representation learning primarily is concerned with word representation learning (Mikolov et al., 2013a,b; Pennington et al., 2014; Goldberg and Levy, 2014; Schwartz et al., 2015; Bojanowski et al., 2017). However, words do not hold the same information content across languages (Section 5.3). Especially in morphologically-rich languages such as Korean, Finnish or Tamil, one word can express the same meaning as several words in English. Inherently, this means that in morphologically-rich languages we find a higher amount of infrequent words, leading to an increased amount of data sparsity issues. It is therefore crucial to a) measure the impact of such typological and morphological differences on standard modeling frameworks, as well as b) aim to find solutions which can work well across the whole variety of the world's languages.

As discussed in background section I, as well as previous Chapters 3 and 4, word representation development and evaluation largely relies on intrinsic evaluation sets for guidance. Often these are expensive and labour-intensive to produce, as they are created through surveys involving hundreds of annotators (Chapter 3). Although recently there has been an increased effort to produce intrinsic evaluation sets for more languages (Leviant and Reichart, 2015; Camacho-Collados et al., 2015), to the best of our knowledge benchmarking data sets across a very large number of typologically diverse languages still do not exist. Here we are aiming to analyse cross-linguistic factors on a large scale (50 typologically-diverse languages), and therefore choose to evaluate the task of word-level language modeling for analysis. [37]

Language Modeling is a major NLP task, as language model architectures can be implicitly or explicitly contained in a variety of higher-level NLP tasks such as speech recognition (Mikolov et al., 2010), text summarisation (Filippova et al., 2015; Rush et al., 2015) or sequence tagging (Rei, 2017). Its importance has been strengthened recently due to the advent of *context-aware* representational models such as ELMO (Peters et al., 2018) and BERT (Devlin et al., 2019), which operate on the sentence-level, and use a LM objective

[37] Note that relations captured by language models are different from the well-defined relations of similarity or lexical entailment.

for optimization. Context-aware architectures have been found to outperform word representations as a base layer for task-specific models in a large variety of settings (Howard and Ruder, 2018b).

These recent developments further stress the importance of enabling and improving language modeling across a wider array of languages. Even these newer types of models might still contain word representations. For instance, in the case of the skip-gram or CBOW word representation model in *word2vec* (Mikolov et al., 2013a,b), there are word representations both at the *input* of the model (word representations), as well as at the *output* (i.e. word *prediction* part) of the model (context representations). In the case of FastText (Bojanowski et al., 2017), the word representations on the input side have been substituted with an architecture that constructs word representations on the fly from character n-gram representations, thereby making the model character-aware and much better suited to work across a variety of languages. Yet, the word representations at the output side of the model remain. A similar development has taken place in the area of language modeling. A popular language model by Kim et al. (2016), the *CharCNN-LSTM*[38] uses a convolutional neural network (CNN) architecture (LeCun et al., 1990) at the input side of the model to create character-aware representations. The output side of the model, used for its **next-word prediction**, still trains word representations. Especially when considering morphologically-rich languages with their data sparsity issues, it is crucial to take the effect of these word-specific parameters into account.

In this chapter we present a benchmark of the common LM architectures across a typologically-diverse set of 50 languages, as well as demonstrating the positive effect of injecting subword information into the word vectors for next-word prediction. We find the method works especially well for morphologically-rich languages, where training successful word representations is a challenge due to data sparsity issues.

[38]CharCNN-LSTM is one of the key building blocks of ELMO (Peters et al., 2018)

The material in this part has been published in the following papers:

- **Daniela Gerz**, Ivan Vulić, Edoardo Maria Ponti, Roi Reichart, and Anna Korhonen. "On the Relation between Linguistic Typology and (Limitations of) Multilingual Language Modeling." *EMNLP 2018*.
- **Daniela Gerz**, Ivan Vulić, Edoardo Maria Ponti, Jason Naradowsky, Roi Reichart, and Anna Korhonen. "Language modeling for morphologically rich languages: Character-aware modeling for word-level prediction." *Transactions of the Association of Computational Linguistics 6, 451-465*. 2018.

The methodology has been further validated in the following publication:

- Ehsan Shareghi, **Daniela Gerz**, Ivan Vulić, Anna Korhonen. "Show Some Love to Your *n*-grams: A Bit of Progress and Stronger *n*-gram Language Modeling Baselines." *NAACL-HLT 2019*

Section 5.3 in particular is thanks to the contribution of Edoardo Maria Ponti.

# Chapter 5

# Character-Aware Next-Word Prediction

A key challenge in cross-lingual NLP is developing general language-independent architectures. However, this ambition is largely hampered by the variation in structural and semantic properties, i.e. the typological profiles of the world's languages. Here we analyse the implications of this variation on the language modeling (LM) task. We present a large-scale study of state-of-the art *n-gram based* and *neural* language models on 50 typologically diverse languages covering a wide variety of morphological systems. Operating in the *full vocabulary LM setup* focused on *word-level prediction*, we demonstrate that a coarse typology of morphological systems is predictive of absolute LM performance. Moreover, *fine-grained typological features* such as exponence, flexivity, fusion, and inflectional synthesis are borne out to be responsible for the proliferation of low-frequency phenomena which are organically difficult to model by statistical architectures, or for the meaning ambiguity of character n-grams. However, word-level prediction is typically agnostic of such subword-level information (characters and character n-grams) and operates over a closed vocabulary, consisting of a limited word set. Indeed, while subword-aware models boost performance across a variety of NLP tasks, previous work did not evaluate the ability of these models to assist next-word prediction in language modeling tasks. Such subword-level informed models should be particularly effective for morphologically-rich languages (MRLs) that exhibit high type-to-token ratios. We present a novel method for injecting subword-level information into semantic word vectors, integrated into the neural language modeling training, to facilitate word-level prediction. We conduct experiments in the LM setting where the number of infrequent words is large, and demonstrate strong perplexity gains across our 50 languages, especially for morphologically-rich languages. Our study strongly suggests that these features have to be taken into consideration during the construction of next-level language-agnostic LM architectures, capable of handling morphologically complex languages such as Tamil or Korean.

## 5.1 Introduction

Word representations and deep learning has allowed NLP algorithms to dispose of manually-crafted features, and to virtually achieve language independence. However, their performance still varies noticeably across languages due to different underlying data distributions (Bender, 2013; O'Horan et al., 2016). Linguistic typology, the systematic comparison of the world's languages, holds promise to explain these idiosyncrasies and interpret statistical models in terms of variation in language structures (Ponti et al., 2017).

In order to evaluate how cross-lingual structural variation hinders the design of effective general-purpose algorithms, we propose the task of language modeling (LM) as a testbed. In particular, we opt for a full-vocabulary setup where no word encountered at training time is treated as an unknown symbol, in order to **a)** ensure a fair comparison across languages with different word frequency rates and **b)** avoid setting an arbitrary threshold on vocabulary size (Cotterell et al., 2018).

Although there recently has been a tendency towards expanding test language samples, the datasets considered in previous works (Botha and Blunsom, 2014; Vania and Lopez, 2017; Kawakami et al., 2017; Cotterell et al., 2018) are not entirely adequate yet to represent the typological variation and to ground cross-lingual generalisations empirically. Hence, we test several LM architectures (including n-gram, neural, and character-aware models) on a novel and wider set of 50 languages sampled according to stratification principles.

Through this large-scale multilingual analysis, we shed new light on the current limitations of standard LM models and offer support to further developments in multilingual NLP. In particular, we demonstrate that the previous fixed-vocabulary assumption in fact ignores the limitations of language modeling for morphologically rich languages. Moreover, we find a strong correlation across the board between LM model performances and the type of morphological system adopted in each language.

To motivate this correlation we show how fine-grained typological properties interact with the frequency distribution (Zipf, 1949) by regulating word boundaries and the proliferation of word forms; and 2) with the mapping between morphemes (here intended as character n-grams) and meaning, by possibly blurring it.

The chapter is organised as follows. After providing a short overview of multilingual LM and its possible setups (Section 5.2), we describe the cross-lingual variation in morphological systems and propose a novel typologically diverse dataset for LM in Section 5.3. We outline the data in Section 5.4 and benchmarked language models in Section 5.9. Finally, we discuss the results in light of linguistic typology in Section 5.10. Further, we present modeling approaches targeting morphologically-rich languages in particular.

A traditional recurrent neural network (RNN) LM setup operates on a limited closed vocabulary of words (Bengio et al., 2003; Mikolov et al., 2010). The limitation arises due to the model learning parameters exclusive to single words. A standard training procedure for neural LMs *gradually* modifies the parameters based on contextual/distributional information: each occurrence of a word token in training data contributes to the estimate of a word vector (i.e., model parameters) assigned to this word type. Low-frequency words therefore often have incorrect estimates, not having moved far from their random initialisation. A common strategy for dealing with this issue is to simply exclude the low-quality parameters from the model (i.e., to replace them with the <*unk*> placeholder), leading to only a subset of the vocabulary being represented by the model.

This limited vocabulary assumption enables the model to bypass the problem of unreliable word estimates for low-frequency and unseen words, but it does not resolve it. The assumption is far from ideal, partly due to the Zipfian nature of each language (Zipf, 1949), and its limitation is even more pronounced for morphologically-rich languages (MRLs): these languages inherently generate a plethora of words by their morphological systems. As a consequence, there will be a large number of words for which a standard RNN LM cannot guarantee a reliable word estimate.

Since gradual parameter estimation based on contextual information is not feasible for rare phenomena in the *full vocabulary setup* (Adams et al., 2017), it is of crucial importance to construct and enable techniques that can obtain these parameters in alternative ways. One solution is to draw information from additional sources, such as characters and character sequences. As a consequence, such character-aware models should facilitate LM word-level prediction in a real-life LM setup which deals with a large amount of low-frequency or unseen words.

Efforts into this direction have yielded exciting results, primarily on the *input* side of neural LMs. A standard RNN LM architecture relies on two word representation matrices learned during training for its *input* and *next-word* prediction. This effectively means that there are two sets of per-word specific parameters that need to be trained. Recent work shows that it is possible to generate a word representation on-the-fly based on its constituent characters, thereby effectively solving the problem for the parameter set on the *input* side of the model (Kim et al., 2016; Luong and Manning, 2016; Miyamoto and Cho, 2016; Ling et al., 2015). However, it is not straightforward how to advance these ideas to the *output* side of the model, as this second set of word-specific parameters is directly responsible for the next-word prediction: it has to encode a much wider range of information, such as topical and semantic knowledge about words, which cannot be easily obtained from its characters alone (Jozefowicz et al., 2016).

While one solution is to directly output characters instead of words (Graves, 2013; Miyamoto and Cho, 2016), a recent work from Jozefowicz et al. (2016) suggests that such purely character-based architectures, which do not reserve parameters for information specific to single words, cannot attain state-of-the-art LM performance on word-level prediction.

In this chapter, we *combine* the two worlds and propose a novel LM approach which relies on both word-level (i.e., contextual) and subword-level knowledge. In addition to training word-specific parameters for word-level prediction using a regular LM objective, our method encourages the parameters to also reflect subword-level patterns by injecting knowledge about morphology. This information is extracted in an unsupervised manner based on already available information in convolutional filters from earlier network layers. The proposed method leads to large improvements in perplexity across a wide spectrum of languages: 22 in English, 144 in Hebrew, 378 in Finnish, 957 in Korean on our LM benchmarks. We also show that the gains extend to another multilingual LM evaluation set, compiled recently for 7 languages by Kawakami et al. (2017).

We conduct a systematic LM study on 50 typologically diverse languages, sampled to represent a variety of morphological systems. We discuss the implications of typological diversity on the LM task, both theoretically in Section 5.3, and empirically in Section 5.10; we find a clear correspondence between performance of state-of-the art LMs and structural linguistic properties. Further, the consistent perplexity gains across the large sample of languages suggest wide applicability of our novel method.

Finally, this chapter can also be read as a comprehensive multilingual analysis of current LM architectures on a set of languages which is much larger than the ones used in recent LM work (Botha and Blunsom, 2014; Vania and Lopez, 2017; Kawakami et al., 2017). We hope that this article with its new datasets, methodology and models, all available online at http://people.ds.cam.ac.uk/dsg40/lmmrl.html, will pave the way for true multilingual research in language modeling.

## 5.2 Multilingual Language Modeling

**Fixed vs Full Vocabulary Setup.** A majority of word-level language models rely on the fixed-vocabulary assumption: they use a special symbol <UNK> that represents all words not present in the fixed vocabulary $V$, which are termed out-of-vocabulary (OOV). Selecting the set $V$ typically slips under the radar, and can be seen as "something of a black art" despite its enormous impact on final LM performance (Cotterell et al., 2018).[39] Standard LM setups

[39]For instance, Vania and Lopez (2017) report perplexity scores of ≈20 for Finnish when $V$ is fixed to the $5k$ most frequent words. The same model in the full-vocabulary setup obtains perplexity scores of ≈2,000.

either fix the vocabulary $V$ to the top $n$ most frequent words, typically with $n = 10,000$ or $n = 5,000$ (Mikolov et al., 2010; Ling et al., 2015; Vania and Lopez, 2017; Lee et al., 2017, *inter alia*), or include in $V$ only words with a frequency above a certain threshold (typically 2 or 5) (Heafield et al., 2013).

The rationale behind fixing the set $V$ is **a)** to make the language model more robust to handling OOVs and to effectively bypass the problem of unreliable word estimates for low-frequency and unseen words (by ignoring them), and **b)** to enable direct comparisons of absolute perplexity scores across different models. However, this posits a critical challenge as cross-linguistic evaluation becomes uneven. In fact, we witness a larger proportion of vocabulary words replaced by <UNK> in morphologically rich languages because of their higher OOV rates (see Table 5.7). What is more, while the fixed-vocabulary assumption artificially improves the perplexity measure, it actually makes the models less useful, especially in morphologically rich languages, as exemplified in Table 5.1.

| | |
|---|---|
| FI | *Kreikkalaiset sijoittivat geometrian synnyn muinaiseen Egyptiin , jossa sitä tarvittiin maanmittaukseen .* |
| FI (MIN-5) | *<UNK> <UNK> <UNK> synnyn <UNK> Egyptiin , jossa sitä tarvittiin <UNK> .* |
| FI (10K) | *<UNK> <UNK> <UNK> <UNK> <UNK> <UNK> , jossa sitä <UNK> <UNK> .* |
| KO | 그 뒤 한시 백일장에서 장원하여 신동으로 알려졌다. 그러나 그의 집은 지독하게 가난했다 |
| KO (MIN-5) | 그 뒤 <UNK> <UNK> <UNK> <UNK> 알려졌다 . 그러나 그의 집은 <UNK> <UNK> |
| KO (10K) | 그 뒤 <UNK> <UNK> <UNK> <UNK> 알려졌다 그러나 그의 <UNK> <UNK> <UNK> |

Table 5.1 Examples from Finnish and Korean LM datasets after applying the standard fixed-vocabulary assumption. MIN=5: only words with corpus frequency above 5 are retained in the final fixed vocabulary $V$; 10K: $V$ comprises the $10k$ most frequent words.

Our goal is to get a clear picture on how different typological features and the corresponding corpus frequency distributions affect LM performance, without the influence of the unrealistic fixed-vocabulary assumption. Therefore, we work in the *full-vocabulary LM setup* (Adams et al., 2017; Grave et al., 2017). This means that we explicitly decide to retain also infrequent words in the modeled data: $V$ contains all words occurring at least once in the training set, only unseen words from test data are treated as OOVs. We believe that this setup leads to an evaluation that pinpoints the crucial limitations of standard LM architectures.[40]

**Why Not Open Vocabulary Setup?** Recent neural LM architectures have also focused on handling large vocabularies and unseen words using character-aware modeling (Luong and Manning, 2016; Jozefowicz et al., 2016; Kawakami et al., 2017, *inter alia*). This setup

[40]For instance, as discussed later in §5.3 and validated empirically in §5.10, the vocabularies of morphologically-rich languages are inherently larger: it is simply more difficult to learn and make word-level LM predictions in such languages.

is commonly referred to as the *open-vocabulary* setup. However, two distinct approaches with crucial modeling differences are referred to by the same term in the literature. **a)** *Word-level generation* constructs word vectors for arbitrary words from constituent subword-level components, but word-level prediction is still evaluated based on the fixed-vocabulary assumption. **b)** *Character-level generation* predicts characters instead of words.

Given that character-level prediction and word-level prediction operate on entirely different sets of symbols, their performance is hardly comparable. Still, Jozefowicz et al. (2016) report that, in a hybrid setup which evaluates character-level prediction based on word-level perplexity with the fixed-vocabulary assumption, current state-of-the-art word-level prediction models (i.e., the ones we discuss in Section 5.5) still significantly outperform such hybrid character-level prediction approaches. Therefore, we operate in the full-vocabulary setup.

## 5.3 Typology of Morphological Systems

Aiming for a comprehensive multilingual LM evaluation in this study, we survey all possible types of morphological systems (Haspelmath and Sims, 2013), which possibly lead to different performances. Traditionally, languages have been grouped into the four main categories: *isolating*, *fusional*, *introflexive* and *agglutinative*, based on their position along a spectrum measuring the preference on breaking up concepts in many words (on one extreme) or rather compose them into single words (on the other extreme).

The mono-dimensionality of this spectrum has recently been challenged as languages exhibit a multitude of morphological features that do not co-vary across languages (Plank, 2017; Ponti et al., 2018a). The typological database WALS (Dryer and Haspelmath, 2013) documents several of them that are relevant for LM: *inflectional synthesis*, *fusion*, *exponence*, and *flexivity*. Note that the prototypes of traditional categories can be approximated in terms of these features, as shown in Table 5.2, although more combinations are possible.

| Type | Fusion | Exponence | Flexivity | Synthesis |
|---|---|---|---|---|
| Isolating | low | 1:1 | 1:1 | low |
| Fusional | mid | many:1 | 1:many | mid |
| Introflexive | high | many:1 | 1:many | mid |
| Agglutinative | mid | 1:1 | 1:1 | high |

Table 5.2 Traditional morphological types described in terms of selected features from WALS.

Languages specify different subsets of grammatical categories (such as tense for verbs, or number for nouns), and for each category different values are available in each language: for instance, Finnish has less tense values (it lacks a future), whereas Slovene has more number

values (including a dual) compared to English. The feature **inflectional synthesis** for verbs (Bickel and Nichols, 2013) measures how many categories appear on the maximally inflected verb per language. More available categories enlarge the vocabulary (and consequently the OOV rate) with forms instantiating all possible combinations of their values.

Another crucial aspect is how the available grammatical categories are expressed, which can be described by fusion, exponence, and flexivity. **Fusion** measures the degree of connectedness between a grammatical marker to another word. The marker can be (from lower to higher fusion) a separate word, a clitic, an affix, or can affect the form of the root itself (e.g. an umlaut or a tone).

**Exponence** measures the number of categories (e.g., tense, number) a single morpheme tends to convey. Exponence is separative if one grammatical category is conveyed by one morpheme (1:1), and cumulative if multiple categories are grouped into one morpheme (many:1).

**Flexivity** indicates the possibility that the value of a grammatical category be mapped into different morphological forms (1:many). In other terms, lemmas belonging to the same part-of-speech are divided into inflectional classes (such as declension classes for nouns or conjugation classes for verbs), each characterised by a different paradigm, that is, a different set of value-to-form mappings.

The three last features are illustrated by the examples Ex. (5.1)-Ex. (5.4), all uttering the sentence "*I will guard the doors and I will not open (them)*".[41]

(5.1) *tôi sẽ bảo vệ cửa và tôi sẽ không mở*
I FUT guard door and I FUT NEG open (Vietnamese)

(5.2) *kapı-lar-ı koruy-acağ-ım ve aç-may-acağ-ım*
door-PL-ACC guard-FUT-1SG and open-NEG-FUT-1SG (Turkish)

(5.3) *sorvegli-erò le port-e e non apr-irò*
guard-FUT.1SG DEF door-PL and NEG open-FUT.1SG (Italian)

(5.4) *ʻe-šmor ʻal ha-dʻlat-ót vʻ-lo ʻe-ftach otán*
1SG-guard.FUT on DEF-door-PL and-NEG 1SG-wait.FUT them (Hebrew)

In particular, consider how tense and person are expressed on verbs. Vietnamese in Ex. (5.1) puts two particles *tôi* and *sẽ* before the verb, which are distinct (separate exponence), autonomous from the root (no fusion), and fixed (absence of flexivity). Turkish in Ex. (5.2)

[41] All morphological glosses follow the Leipzig glossing rules, listed at https://www.eva.mpg.de/lingua/resources/glossing-rules.php

attaches suffixes: *-acak-* for tense and *-ım* for person. These are distinct (separate exponence), joined to the roots (concatenative fusion), and (phonologically determined variants of) the same morpheme (1:1 flexivity). Italian in Ex. (5.3) uses affixes *-erò* and *-irò*: they are concatenated to the root with respect to fusion, convey both tense and person (cumulative exponence), and are dissimilar (presence of flexivity). Finally, in Ex. (5.4) for Hebrew the consonant pattern of the verb *š-m-r* is interdigitated by the vowel *-o-* for tense, and preceded by a prefix *ʻe-* for person. The first phenomenon alters the root itself (introflexive fusion), is distinct from the second (separate exponence), and changes its realisation based on the verb's lemma (presence of flexivity).

The above evidence strongly motivates us, as well as recent previous work (Vania and Lopez, 2017; Kawakami et al., 2017; Cotterell et al., 2018), to approach LM with models that are aware of the inner structure of their input words, and to benchmark these modeling choices on a typologically diverse range of languages.

## 5.4 Data

**Selection of Languages.** Our selection of test languages is guided by the following goals: **a)** we have to ensure the coverage of typological properties from Section 5.3, and **b)** we want to analyse a large set of languages which extends and surpasses other work in the LM literature (see Section 2.3.

Since cross-lingual NLP aims at modeling *extant* languages rather than *possible* languages (including, e.g., extinct ones), creating a balanced sample is challenging. In fact, attested languages, intended as a random variable, are extremely sparse and not independent-and-identically-distributed (Cotterell and Eisner, 2017). First, available and reliable data exist only for a fraction of the world's languages. Second, these data sets are biased because their features may not stem from the underlying distribution, i.e., from what is naturally possible/frequent, but rather can be inherited by genealogical relatedness or borrowed by areal proximity (Bakker, 2010). To mitigate these biases, theoretical works resorted to stratification approaches, where each subgroup of related languages is sampled independently. maximizing their diversity (Dryer, 1989, *inter alia*). We perform our selection in the same spirit.

We start from the Polyglot Wikipedia (PW) project (Al-Rfou et al., 2013) which provides cleaned and tokenised Wikipedia data in 40 languages. However, the majority of the PW languages are similar from the perspective of genealogy (26/40 are Indo-European), geography (28/40 are Western European), and typology (26/40 are fusional). Consequently, the PW set is not a representative sample of the world's languages.

To amend this limitation, we source additional languages with the data coming from the same domain, Wikipedia, considering candidates in descending order of corpus size cleaned and preprocessed by the Polyglot tokeniser (Al-Rfou et al., 2013). Since fusional languages are already represented in the PW, we add new languages from other morphological types: isolating (*Min Nan, Burmese, Khmer*), agglutinative (*Basque, Georgian, Kannada, Tamil, Mongolian, Javanese*), and introflexive languages (*Amharic*).

**Partition.** We construct datasets for all 50 languages by extracting the first 40K sentences for each language, and split them into train (34K), validation (3K), and test (3K). This choice has been motivated by the following observations: **a)** we require similarly-sized datasets from the same domain for all languages; **b)** the size of the datasets has to be similar to the standard English PTB dataset (Marcus et al., 1993) which has been utilised to guide LM development in English for more than 20 years. The final list of 50 languages along with their language codes (ISO 639-1), morphological type (i.e., isolating, fusional, introflexive, agglutinative), and corpus statistics is provided in Table 5.7.

## 5.5 (Baseline) Language Models

The availability of LM evaluation sets in a large number of diverse languages, described in Section 5.4, now provides an opportunity to perform a full-fledged multilingual analysis of representative LM architectures. At the same time, these different architectures serve as the baselines for our novel model which fine-tunes the output matrix $M^w$.

As mentioned, the traditional LM setup is to use words both on the input and on the output side (Goodman, 2001; Bengio et al., 2003; Deschacht and Moens, 2009) relying on n-gram word sequences. We evaluate a strong model from the *n-gram* family of models from the KenLM package (*https://github.com/kpu/kenlm*): it is based on 5-grams with extended Kneser-Ney smoothing (**KN5**) (Kneser and Ney, 1995; Heafield et al., 2013) [42]. The rationale behind including this *non-neural* model is to also probe the limitations of such n-gram-based LM architectures on a diverse set of languages. [43]

Recurrent neural networks (RNNs), especially Long-Short-Term Memory networks (LSTMs), have taken over the LM universe recently (Mikolov et al., 2010; Sundermeyer et al., 2015; Chen et al., 2016, i.a.). These LMs map a sequence of input words to embedding vectors using a look-up matrix. The embeddings are passed to the LSTM as input, and the

[42] We evaluate the default setup for this model using the option `-interpolate_unigrams=1` which avoids assigning zero-probability to unseen words.

[43] This work has since been extended in collaboration with Shareghi et al. (2019).

model is trained in an autoregressive fashion to predict the next word from the pre-defined vocabulary given the current context. As a strong baseline from this LM family, we train a standard LSTM LM (**LSTM-Word**) relying on the setup from Zaremba et al. (2015) (see Table 5.5).

Finally, we also evaluate a character-aware variant of the neural LSTM LM architecture. We use the **Char-CNN-LSTM** model (Kim et al., 2016) due to its public availability and strong performance in several languages. In this model, each character is embedded and passed through a convolutional neural network with max-over-time pooling (LeCun et al., 1990), followed by a highway network transformation (Srivastava et al., 2015) to build word representations from their constituent characters. By resorting to character-level information, the model is able to provide better parameter estimates for lower-frequency words, which is particularly important for morphologically rich languages. The CNN-based word representations are then processed in a sequence by a regular LSTM network to obtain word-level predictions.

## 5.6 Underlying LM: Char-CNN-LSTM

As the underlying model we opt for the state-of-the-art neural LM architecture of Kim et al. (2016): it has been shown to work across a number of languages and in a large-scale setup (Jozefowicz et al., 2016). It already provides a solution for the *input* side parameters of the model by building word vectors based on the word's constituent character sequences. However, its *output* side still operates with a standard word-level matrix within the closed and limited vocabulary assumption. We refer to this model as **Char-CNN-LSTM** and describe its details in the following. Figure 5.1 (left) illustrates the model architecture.

Char-CNN-LSTM constructs input word vectors based on the characters in each word using a convolutional neural network (CNN) (LeCun et al., 1990), then processes the input word-level using a LSTM (Hochreiter and Schmidhuber, 1997). The next word is predicted using word embeddings, a large number of parameters which have to be trained specifically to represent the semantics of single words. We refer to this space of word representations as $M^w$.

Formally, for the input layer the model trains a look-up matrix $C \in \mathbb{R}^{|V^c| \times d_c}$, corresponding to one $d_c$-dimensional vector per character $c$ in the char vocabulary $V^c$. For each input, it takes a sequence of characters of a fixed length $m$, $[c_1, ...c_m]$, where $m$ is the maximum length of all words in the word vocabulary $V^w$, and the length of each word is $l \leq m$.

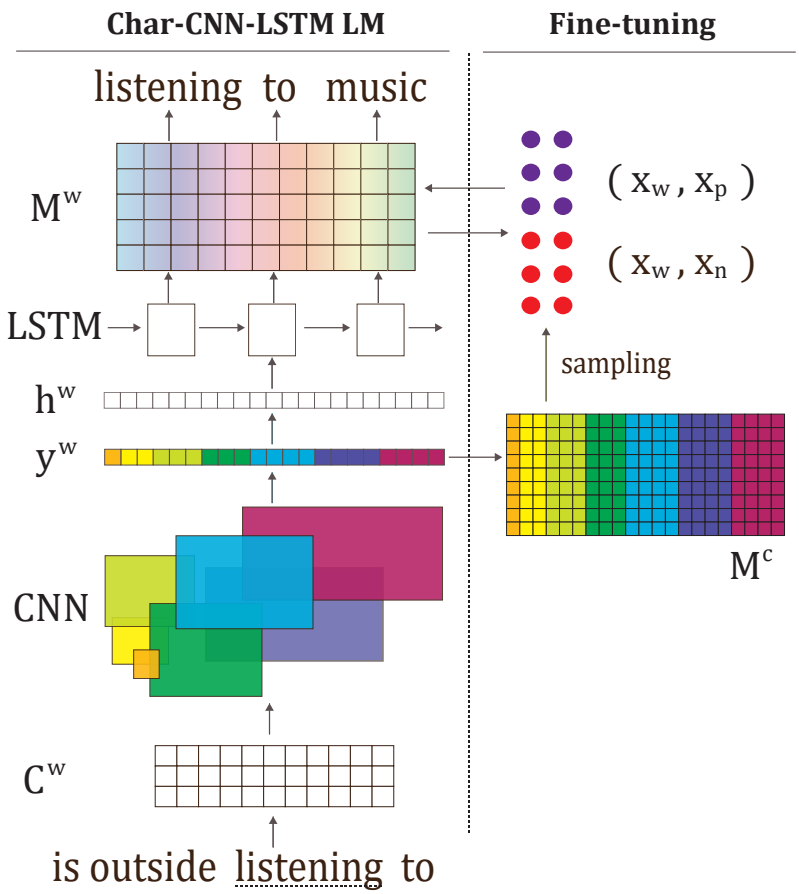



Figure 5.1 An illustration of the Char-CNN-LSTM LM and our fine-tuning post-processing method. After each epoch we adapt word-level vectors in the softmax embedding $M^w$ using samples based on features from the char-level convolutional filters. The figure follows the model flow bottom to the top.

Looking up all characters of a word yields a sequence of char representations in $\mathbb{R}^{d_c \times l}$, which is zero-padded to fit the fixed length $m$. For each word one gets a sequence of char representations $C^w \in \mathbb{R}^{d_c \times m}$, passed through a 1D convolution:

$$f_i^w = tanh(\langle C^w, H_i \rangle + b). \tag{5.5}$$

$H_i \in \mathbb{R}^{d_{f,i} \times s_i}$ is a *filter* or *kernel* of size/width $s_i$ and $\langle A, B \rangle = Tr(AB^T)$ is the Frobenius inner product. The model has multiple filters, $H_i$, with kernels of different width, $s_i$, and dimensionality $d_{f,i}$, $i$ is used to index filters. Since the model performs a convolution over char embeddings, $s_i$ corresponds to the char window the convolution is operating on: e.g.,

a filter of width $s_i = 3$ and $d_{3,i} = 150$ could be seen as learning 150 features for detecting 3-grams.

By learning kernels of different width, $s_i$, the model can learn subword-level features for character sequences of different lengths. $f_i^w$ is the output of taking the convolution with filter $H_i$ for word $w$. Since $f_i^w$ can get quite large, its dimensionality is reduced using *max-over-time* (1D) pooling: $y_i^w = \max_j f_i^w[j]$. Here, $j$ indexes the dimensions $d_{f,i}$ of the filter $f_i^w$, and $y_i^w \in \mathbb{R}^{d_{f,i}}$. This corresponds to taking the maximum value for each feature of $H_i$, with the intuition that the most informative feature would have the highest activation. The output of all max-pooling operations $y_i^w$ is concatenated to form a word vector $y^w \in \mathbb{R}^{d_p}$, where $d_p$ is the number of all features for all $H_i$:

$$y^w = concat([y_1^w, ..., y_i^w]). \tag{5.6}$$

This vector is passed through a highway network (Srivastava et al., 2015) to give the network the possibility to reweigh or transform the features: $h^w = Highway(y^w)$.[44] So far all transformations were done per word; after the highway transformation word representations are processed in a sequence by an LSTM (Hochreiter and Schmidhuber, 1997):

$$o_t^w = LSTM([h_{w_1}, ...h_{w_{t-1}}]). \tag{5.7}$$

The LSTM yields one output vector $o_t^w$ per word in the sequence, given all previous time steps $[y_{w_1}, ...y_{w_{t-1}}]$. To predict the next word $w_{t+1}$, one takes the dot product of the vector $o_t^w \in \mathbb{R}^{1\times d_l}$ with a lookup matrix $M^w \in \mathbb{R}^{d_l \times |V^w|}$, where $d_l$ corresponds to the LSTM hidden state size. The vector $p_{t+1} \in \mathbb{R}^{1\times|V^w|}$ is normalised to contain values between 0 and 1, representing a probability distribution over the next word. This corresponds to calculating the softmax function for every word $k$ in $V^w$:

$$p(w_{t+1} = k|o_t) = \frac{e^{(o_t \cdot m_k)}}{\sum_{k' \in V_w} e^{(o_t \cdot m_{k'})}} \tag{5.8}$$

where $P(w_{t+1} = k|o_t)$ is the probability of the next word $w_{t+1}$ being $k$ given $o_t$, and $m_k$ is the output embedding vector taken from $M^w$.

**Word-Level Vector Space: $M^w$** The model parameters in $M^w$ can be seen as the bottleneck of the model, as they need to be trained specifically for single words, leading to unreliable estimates for infrequent words. As an analysis of the corpus statistics later in Section 5.10

[44] We adopt this part from Kim et al. (2016). The highway network gives about 1-2% better performance on this task in our experience.

reveals, the Zipfian effect and its influence on word vector estimation cannot be fully resolved even with a large corpus, especially taking into account how flexible MRLs are in terms of word formation and combination. Yet, having a good estimate for the parameters in $M^w$ is essential for the final LM performance, as they are directly responsible for the next-word prediction.

Therefore, our aim is to improve the quality of representations in $M^w$, focusing on infrequent words. To achieve this, we turn to another source of information: character patterns. In other words, since $M^w$ does not have any information about character patterns from lower layers, we seek a way to: **a)** detect words with similar subword structures (i.e., "morpheme"-level information), and **b)** let these words share their semantic information.

## 5.7 Character-Aware Vector Space

The CNN part of CharCNN-LSTM, see Eq. (5.6), in fact provides information about such subword-level patterns: the model constructs a word vector $y^w$ on-the-fly based on the word's constituent characters. We let the model construct $y^w$ for all words in the vocabulary, resulting in a *character-aware word vector space* $M^c \in \mathbb{R}^{|V^w| \times d_p}$. The construction of the space is completely unsupervised and independent of the word's context; only the first (CNN) network layers are activated. Our core idea is to leverage this information obtained from $M^c$ to influence the output matrix $M^w$, and consequently the network prediction, and extend the model to handle unseen words.

Fine-tuning with information from within the model is helpful in this case because the model learns complementary information in both matrices: $M^c$ learns orthographic similarity, while $M^w$ learns semantic or syntactic similarity. Low-frequency words will not have a well-trained vector in $M^w$, therefore our approach allows the model to "*guess*" the meaning of a low-frequency word based on its orthography. Intuitively, this is similar to how a human would deal with an unknown word: taking a guess based on its orthographic similarity to known words. This is especially helpful for morphologically-rich languages, due to their large number of infrequent words and highly structured orthography.

We now first take a closer look at the character-aware space $M^c$, and then describe how to improve and expand the semantic space $M^w$ based on the information contained in $M^c$ (Section 5.8). Each vocabulary entry in $M^c$ encodes character n-gram patterns about the represented word, for $1 > n \leq 7$. The n-gram patterns arise through filters of different lengths, and their maximum activation is concatenated to form each individual vector $y^w$. The matrix $M^c$ is of dimensionality $|V^w| \times 1100$, where each of the 1,100 dimensions corresponds to the activation of one kernel feature. In practice, dimensions $[0, 1, .. : 50]$ correspond to single-

character features, [50 : 150] to character 2-grams, [150 : 300] to 3-grams. The higher-order $n$-grams get assigned 200 dimensions each, up to dimensions [900 : 1100] for 7-grams.

| | $s_i$ | Pattern | Max Activations |
|---|---|---|---|
| ZH | 1 | 更, 不 | **更**为, **更**改, **更**名, .., **不**满, **不**明, **不**易 |
| | 1 | 今, 代 | **今**日, **今**人, 至少, .., 如何, 现**代**, 当**代** |
| TR | 1 | Caps | **I**n, **E**bru, **VIC**,..,**FAT**, **MW**, **MIT** |
| | 3 | mu- | .., **mu**tfağının, **mu**harebe, **mu**htelif |
| | 6 | Üniversite | .., **Üniversite**si'nin, **üniversite**lerde |

Table 5.3 Each CNN filter tends to have high activations for a small number of subword patterns. $s_i$ denotes the filter size.

Drawing an analogy to work in computer vision (Zeiler and Fergus, 2014; Chatfield et al., 2014), we delve deeper into the filter activations and analyse the key properties of the vector space $M^c$. The qualitative analysis reveals that many features are interpretable by humans, and indeed correspond to frequent subword patterns, as illustrated in Table 5.3. For instance, tokenised Chinese data favours short words: consequently short filters activate strongly for one or two characters. The first two filters (width 1) are highly active for two common single characters each: one filter is active for 更 *(again, more)*, 不 *(not)*, and the other for 今 *(now)*, 代 *(time period)*. Larger filters (width 5-7) do not show interpretable patterns in Chinese, since the vocabulary largely consists of short words (length 1-4).

Agglutinative languages show a tendency towards long words. We find that medium-sized filters (width 3-5) are active for morphemes or short common subword units, and the long filters are activated for different surface realisations of the same root word. In Turkish, one filter is highly active on various forms of the word *üniversite (university)*. Further, in MRLs with the Latin alphabet short filters are typically active on capitalisation or special chars.

| | Word | Nearest Neighbours |
|---|---|---|
| DE | Ursprünglichkeit | **ursprünglich**e, **Ur**stoff, **ursprünglich**en |
| | Mittelwert | **Mittelwert**en,Regel**werk**es,**Mittel**weser |
| | effektiv | **Effekt**,Perf**ekt**,**Effekt**e,p**erfekt**en,R**espekt** |
| JA | 大学 | 大金, 大石, 大震災, 大空, 大野 |
| | ハイク | ハイム , バイク, メイク, ハッサク |
| | 1725 | **18**2**5**, **16**2**5**, **15**24mm, **17**28 |
| | Magenta | **Ma**plet, **Ma**ya, **Ma**nagement |

Table 5.4 Nearest neighbours for vocabulary words, based on the character-aware vector space $M^c$.

Table 5.4 shows examples of nearest neighbours based on the activations in $M^c$. The space seems to be arranged according to shared subword patterns based on the CNN features. It does not rely only on a simple character overlap, but also captures shared morphemes. This

property is exploited to influence the LM output word embedding matrix $M^w$ in a completely unsupervised way, as illustrated on the right side of Figure 5.1.

## 5.8 Fine-Tuning the LM Prediction

While the output vector space $M^w$ captures word-level semantics, $M^c$ arranges words by subword features. A model which relies solely on character-level knowledge (similar to the information stored in $M^c$) for word-level prediction cannot fully capture word-level semantics and even hurts LM performance (Jozefowicz et al., 2016). However, shared subword units still provide useful evidence of shared semantics (Cotterell et al., 2016; Vulić et al., 2017): injecting this into the space $M^w$ to *additionally* reflect shared subword-level information should lead to improved word vector estimates, especially for MRLs.

### 5.8.1 Fine-Tuning and Constraints

We inject this information into $M^w$ by adapting recent fine-tuning (often termed *retrofitting* or *specialisation*) methods for vector space post-processing (Faruqui et al., 2015; Wieting et al., 2015; Mrkšić et al., 2017; Vulić et al., 2017, i.a.). These models enrich initial vector spaces by encoding external knowledge provided in the form of simple *linguistic constraints* (i.e., word pairs) into the initial vector space.

There are two fundamental differences between our work and previous work on specialisation. First, previous models typically use rich hand-crafted lexical resources such as WordNet (Fellbaum, 1998) or the Paraphrase Database (Ganitkevitch et al., 2013), or manually defined rules (Vulić et al., 2017) to extract the constraints, while we generate them directly using the implicit knowledge coded in $M^c$. Second, our method is *integrated* into a language model: it performs updates after each epoch of the LM training.[45] In Section 5.8.2, we describe our model for fine-tuning $M^w$ based on the information provided in $M^c$.

Our fine-tuning approach relies on constraints: positive and negative word pairs $(x_i, x_j)$, where $x_i, x_j \in V^w$. Iterating over each *cue word* $x_w \in V^w$ we find a set of positive word pairs $P_w$ and negative word pairs $N_w$: their extraction is based on their (dis)similarity with $x_w$ in $M^c$. Positive pairs $(x_w, x_p)$ contain words $x_p$ yielding the highest cosine similarity to the $x_w$ (=nearest neighbors) in $M^c$. Negative pairs $(x_w, x_n)$ are constructed by randomly sampling words $x_n$ from the vocabulary. Since $M^c$ gets updated during the LM training, we (re)generate the sets $P_w$ and $N_w$ after each epoch.

[45] We have also experimented with a variant which performs only a post-hoc single update of the $M^w$ matrix after the LM training, but a variant which performs continuous per-epoch updates is more beneficial for the final LM performance.

### 5.8.2 Attract-Preserve

We now present a method for fine-tuning the output matrix $M^w$ within the CharCNN-LSTM LM framework. As said, the fine-tuning procedure runs after each epoch of the standard log-likelihood LM training (see Figure 5.1). We adapt a variant of a state-of-the-art post-processing specialisation procedure (Wieting et al., 2015; Mrkšić et al., 2017). The idea of the fine-tuning method, which we label **Attract-Preserve (AP)**, is to pull the positive pairs closer together in the output word-level space, while pushing the negative pairs further away.

Let $v_i$ denote the word vector of the word $x_i$. The AP cost function has two parts: *attract* and *preserve*. In the *attract* term, using the extracted sets $P_w$ and $N_w$, we push the vector of $x_w$ to be closer to $x_p$ by a similarity margin $\delta$ than to its negative sample $x_n$:

$$attr(P_w, N_w) = \sum_{\substack{(x_w, x_p) \in P_w, \\ (x_w, x_n) \in N_w}} ReLU(\delta + v_w v_n - v_w v_p).$$

$ReLU(x)$ is the standard rectified linear unit (Nair and Hinton, 2010). The $\delta$ margin is set to 0.6 in all experiments as in prior work (Mrkšić et al., 2017) without any subsequent fine-tuning.

The preserve cost acts as a regularisation pulling the "fine-tuned" vector back to its initial value:

$$pres(P_w, N_w) = \sum_{x_w \in V^w} \lambda_{reg} ||\hat{v}_w - v_w||_2. \tag{5.9}$$

$\lambda_{reg} = 10^{-9}$ is the $L_2$-regularisation constant (Mrkšić et al., 2017); $\hat{v}_w$ is the original word vector before the procedure. This term tries to preserve the semantic content present in the original vector space, as long as this information does not contradict the knowledge injected by the constraints. The final cost function adds the two costs: $cost = attr + pres$.

## 5.9 Experimental Setup

**Evaluation Setup** We report *perplexity* scores (Jurafsky and Martin, 2017, Chapter 2.4, Chapter 4.2.1) using the *full* vocabulary of the respective LM dataset. For n-gram language models this is the de facto standard setup, as n-gram language models provide a stringent way of dealing with out-of-vocabulary (OOV) and rare words without relying on any pruning (Heafield et al., 2013; Shareghi et al., 2019). However, in neural LMs this remains an open question (Kawakami et al., 2017; Kim et al., 2016; Jozefowicz et al., 2016). A common practice for neural LMs is pruning the training corpus and imposing a *closed vocabulary*

assumption (Mikolov et al. (2010)) where rare words at training and unseen words at test are treated as an <*unk*> token.

Motivated by the observation that infrequent words constitute a significant part of the vocabulary in MRLs, and that vocabulary sizes naturally differ between languages, we have decided to avoid the <*unk*> placeholder for low-frequency words, and run all neural models on the full vocabulary (Adams et al., 2017; Grave et al., 2017).

In our setup the vocabulary contains all words occurring at least once in the training set. To ensure a fair comparison between all neural models, words occurring only in the test set are mapped to a random vector with the same technique for all neural models, as described next.

**Sampling Vectors of Unseen Words** Since zero-shot semantic vector estimation at test time is an unresolved problem, we seek an alternative way to compare model predictions at test time. We report all results with unseen test words being mapped to *one* randomly sampled <*unk*> vector. The <*unk*> vector is part of the vocabulary at training time, but remains untrained and at its random initialization since it never occurs in the training data. Therefore, we sample a random <*unk*> vector at test time from the same part of the space as the trained vectors, using a normal distribution with the mean and the variance of $M^w$ and the same fixed random seed for all models. We employ this methodology for *all* neural LM models, and thereby ensure that results are comparable.

**Unseen and Rare Words in N-gram vs Neural Models** All neural models operate on exactly the same vocabulary and treat out-of-vocabulary (OOV) words in exactly the same way. As mentioned, we include KN5 as a strong (non-neural) baseline to give perspective on how this more traditional model performs across 50 typologically diverse languages. We have selected the setup for the KN5 model to be as close as possible to that of neural LMs. However, due to the different nature of the models, we note that the results between KN5 and other models might be regarded as not comparable.

In KN5 discounts are added for low-frequency words, and unseen words at test time are regarded as outliers and assigned low probability estimates. In contrast, for all neural models we sample unseen word vectors to lie in the space of trained vectors (see before). We find the latter setup to better reflect our intuition that especially in MRLs unseen words are not outliers but often arise due to morphological complexity.

We contributed to a further discussion to comparing n-gram and neural models in Shareghi et al. (2019). N-gram and neural language models have different underlying assumptions regarding rare and unseen words. However, running both in the full vocabulary setup provides

a fairer comparison, as opposed to using the traditional n-gram models in the common closed-vocabulary neural setup. Pruning the vocabulary as common with neural models will discard the discount parameters and therefore unfairly damage the performance of n-gram models (Shareghi et al., 2019). The full vocabulary setup therefore is a better fit for our purposes both considering comparability to n-gram models as well as MRLs.

**Training Setup and Parameters** We reproduce the standard LM setup of Zaremba et al. (2015) and parameter choices of Kim et al. (2016), with batches of 20 and a sequence length of 35, where one step corresponds to one token. The maximum word length is chosen dynamically based on the longest word in the corpus. The corpus is processed continuously, and the RNN hidden state resets occur at the beginning of each epoch. Parameters are optimised with stochastic gradient descent. The gradient is averaged over the batch size and sequence length. We then scale the averaged gradient by the sequence length (=35) and clip to 5.0 for more stable training. The learning rate is 1.0, decayed by 0.5 after each epoch if the validation perplexity does not improve. We train all models for 15 epochs on our 50 training sets extracted from the PW, and for 30 epochs on the MWC and EP corpora, which is typically sufficient for model convergence.

| | |
|---|---|
| Character embedding size | 15 |
| Word embedding size | 650 |
| Number of RNN layers | 2 |
| Number of highway layers | 2 |
| Dropout value | 0.5 |
| Optimizer | SGD |
| Learning rate | 1.0 |
| Learning rate decay | 0.5 |
| Parameter init: rand uniform | [-0.05, 0.05] |
| Batch size | 20 |
| RNN sequence length | 35 |
| Max grad norm | 5.0 |
| Max word length | dynamic |
| Max epochs | 15 or 30 |
| AP margin ($\delta$) | 0.6 |
| AP optimizer | Adagrad |
| AP learning rate | 0.05 |
| AP gradient clip | 2 |
| AP regularization constant | $10^{-9}$ |
| AP rare words frequency threshold | 5 |

Table 5.5 Hyper-parameters.

Our AP fine-tuning method operates on the whole $M^w$ space, but we only allow words more frequent than 5 as cue words $x_w$ (see Section 5.8 again), while there are no restrictions

on $x_p$ and $x_n$.[46] Our preliminary analysis on the influence of the number of nearest neighbours in $M^c$ shows that this parameter has only a moderate effect on the final LM scores. We thus fix it to 3 positive and 3 negative samples for each $x_w$ without any tuning. AP is optimised with Adagrad (Duchi et al., 2011) and a learning rate of 0.05, the gradients are clipped to $\pm 2$.[47] A full summary of all hyper-parameters and their values is provided in Table 5.5.

## 5.10 Results and Discussion

In this section, we present our findings on the connection between LM performance and corpus statistics emerging from different typological profiles (see Section 5.3), as well as analyse the influence of AP fine-tuning. Before proceeding, we stress that the absolute perplexity scores across different languages are not directly comparable, but their values provide evidence on the difficulty and limitations of language modeling in each language, considering the fact that all language models were trained on similarly-sized datasets. The results for all three benchmarked language models on all 50 languages are summarised in Table 5.7.

**Comparison of Baseline Language Models.** A quick inspection of the results from Table 5.7 reveals that the CharCNN-LSTM model is the best-performing baseline model overall. We report the best results with that model for 48/50 languages and across all traditional morphological types. Gains over the simpler recurrent LM architecture (i.e., the LSTM model) are present for all 50/50 languages. In short, this means that character-level information on the input side of neural architectures, in addition to leading to fewer parameters, is universally beneficial for the final performance of word-level prediction, as also suggested by Kim et al. (2016) on a much smaller set of languages. By relying on character-level knowledge, CharCNN-LSTM model provides better estimates for lower-frequency words.

Moreover, the results show that KN5 is a competitive baseline for several languages (e.g., Kannada, Thai, Amharic). This further highlights the importance of testing models on a typologically diverse set of languages: despite the clear superiority of neural LM architectures such as CharCNN-LSTM in a large number of languages, the results and the marked outliers still suggest that there is currently no "one-size-fits-all" model. This finding has been further validated in Shareghi et al. (2019), where we additionally report scores with modern n-

[46] This choice has been motivated by the observation that rare words tend to have other rare words as their nearest neighbours. Note that vectors of words from positive and negative examples, and not only cue words, also get updated by the AP method.

[47] All scores with neural models are produced with our own implementations in TensorFlow (Abadi et al., 2016).

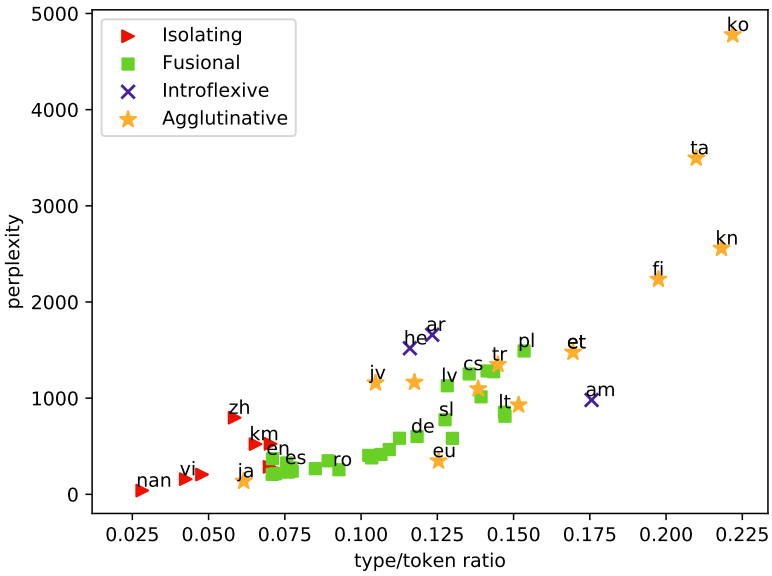



Figure 5.2 Perplexity scores with the CharCNN-LSTM language model (Kim et al., 2016) on PTB-sized language modeling data in 50 languages as a function of type-to-token ratios in training data.

gram modeling approaches, which achieved competitive results especially on challenging morphologically-rich languages.

In general, large perplexity scores for certain languages (e.g., agglutinative languages such as Finnish, Korean, Tamil, or introflexive languages), especially when compared to performance on English on a similarly-sized dataset, clearly point at the limitations of all the "language-agnostic" LM architectures. As suggested by Jozefowicz et al. (2016), LM performance in English can be boosted by simply collecting more data and working with large vocabularies (e.g., reducing the number of relevant OOVs). However, this solution is certainly not applicable to a majority of the world's languages (Bird, 2011; Gandhe et al., 2014; Adams et al., 2017), see later in Section 5.10: *Further Discussion*.

**Frequency Analysis and Traditional Morphological Types.** We now analyse all languages in our collection according to word-level frequency properties also listed in Table 5.7 for all 50 languages. We report: 1) the vocabulary size (i.e., the total number of vocabulary words in each training dataset); 2) the total number of test words not occurring in the corresponding training data; 3) the total number of tokens in both training and test data; and finally 4) type-to-token ratios (TTR) in training data. We also plot absolute perplexity scores

of CharCNN-LSTM (Kim et al., 2016), the best-performing model overall (see Section 5.10), in relation to TTR ratios in Figure 5.2.

In isolating and some fusional languages (e.g., Vietnamese, Thai, English) the TTR tends to be small: we have a comparatively low number of infrequent words. Agglutinative languages such as Finnish, Estonian, and Korean are on the other side of the spectrum. Introflexive and fusional languages, typically over-represented in prior work (see the discussion in Section 5.3), are found in the middle.

This emerges clearly in Figure 5.2, grouping isolating languages to the left side of the x-axis, followed by fusional languages (Germanic and Romance first to the left, and then Balto-Slavic to the right), and placing agglutinative languages towards the far right. Crucially, TTR is an excellent predictor of LM performance. To measure the correlation between this corpus statistics variable and absolute LM performance, we compute their Pearson's $r$ correlation. We find a strong positive correlation, with a value of $r = 0.83$ and significance $p < 0.001$.

We do observe a strong link between each language's morphological type, and the corresponding perplexity score. A transition in terms of the spectrum of morphological systems (see Section 5.3) can be traced again on the y-axis of Figure 5.2, roughly following the reported LM performance: from isolating, over fusional and introflexive to agglutinative languages. In fact, a correlation exists also between traditional morphological types and LM performance. We assessed its strength with the one-way ANOVA statistical test, obtaining a value of $\eta^2 = 0.37$ and a significance of $p < 0.001$.

Finally, it should be noted that the choice of TTP over other corpus statistics such as vocabulary size is motivated by the fact that the corpora are comparable, and not parallel. Because of this, the variation of $V$ may stem from the contents rather than the intrinsic linguistic properties. As a counter-check, the correlation between $V$ and LM performance is in fact milder, with $r = 0.64$. Yet, notwithstanding the stronger correlation, TTP is unable to explain the results entirely. Only through finer-grained typological features it becomes possible to justify several outliers, as shown in the next subsection.

| Variables | | | | | | |
|---|---|---|---|---|---|---|
| Independent | Dependent | Statistical Test | Models | | | |
| | | | KN5 | LSTM | +Char-CNN | ++AP |
| Train type/token | PPL | Pearson's $\rho$ 0.833 | 0.813 | 0.823 | 0.831 | |
| Test new types | PPL | Pearson's $\rho$ | 0.860 | 0.803 | 0.818 | 0.819 |
| Morphology | PPL | one-way ANOVA $\eta^2$ | 0.354 | 0.338 | 0.369 | 0.374 |
| | | | LSTM vs +CharCNN | | +CharCNN vs ++AP | |
| Train type/token | $\Delta$ PPL | Pearson's $\rho$ | 0.729 | | 0.778 | |
| Morphology | $\Delta$ PPL | one-way ANOVA $\eta^2$ | 0.308 | | 0.284 | |

Table 5.6 Correlations between model performance and language typology as well as with corpus statistics (type/token ratio and new word types in test data). All variables are good performance predictions.

**Fine-Grained Typological Analysis.** Among the relevant typological features (see Section 5.3 and Table 5.2), fusion and inflectional synthesis have the largest impact on word-level predictions. In fact, the former determines the word boundaries, whereas the latter regulates the amount of possible morpheme combinations. Consider their effect on the frequency distribution of words, expressed as follows (Zipf, 1949):

$$f = \frac{\frac{1}{k^s}}{\sum_{n=1}^{V} \frac{1}{n^s}} \tag{5.10}$$

$f$ is the frequency, $k$ the rank, and $s \geq 0$ the exponent characteristic of the distribution. If high, both typological features enlarge $V$ and $s$, assigning less probability mass to each word.

Low fusion means a preference for separate words (as in isolating languages such as Vietnamese and Chinese), leading to a smaller vocabulary with less (but more frequent) words. This property, additionally boosted by low inflectional synthesis, facilitates statistical language modeling in isolating languages. Vice versa, high fusion results in preference for concatenation of morphemes or introflection, and consequently sparser vocabularies. Yet, this distinction cannot justify the figures by itself, as it equates agglutinative languages and traditional fusional languages. Here, inflectional synthesis is also at play. Through the statistical test of one-way ANOVA, we found a weak effect of $\eta^2 = 0.09$ for fusion and a medium effect of $\eta^2 = 0.21$ for inflection synthesis.

On the other hand, the fine-grained typological features of exponence and flexivity play a role in the ambiguity of the mapping between morphemes and meanings or grammatical functions. This turns out to be especially relevant for character-aware models. The intuition is that if the mapping is straightforward, injecting character information is more advantageous. To validate this claim, we evaluate the ANOVA between exponence of nouns and verbs and the difference in perplexity between LSTM and CharCNN-LSTM.[48] We report a weak, although existent, correlation with value $\eta^2 = 0.07$ and $\eta^2 = 0.04$, respectively.

**Further Discussion.** Importantly, our large-scale multilingual LM study strongly indicates that due to diverse typological profiles, certain languages and language groups are inherently more complex to language-model when relying on established statistical models, even when such models are constructed as widely applicable and (arguably) language-agnostic. This finding supports preliminary results from prior work (Botha and Blunsom, 2014; Adams et al., 2017; Cotterell et al., 2018), and is also backed by insights from linguistic theory on variance of language complexity in general and variance of morphological complexity in specific (McWhorter, 2001; Evans and Levinson, 2009). More broadly and along the same line, earlier

[48] Unfortunately no values are available in WALS for the feature of flexivity besides a limited domain.

research in statistical machine translation (SMT) has also shown that typological factors such as the amount of reordering, the morphological complexity, as well as genealogical relatedness of languages are crucial in predicting success in SMT (Birch et al., 2008; Paul et al., 2009; Daiber, 2018).

Our results indicate that the artificial fixed-vocabulary assumption from prior work produces overly optimistic perplexity scores, and its limitation is even more pronounced in morphologically rich languages, which inherently contain a large number of infrequent words due to their productive morphological systems. The typical solution to collect more data (Jozefowicz et al., 2016; Kawakami et al., 2017) mitigates this effect to a certain extent, but stills suffers from the Zipfian hypothesis (1949), and it cannot be guaranteed for resource-poor languages where obtaining sufficient monolingual data is also a challenge (Adams et al., 2017).

Therefore, another solution is to resort to other sources of information which are not purely contextual/distributional. For instance, a promising line of current and future research is to (learn to) exploit subword-level patterns captured in an unsupervised manner (Pinter et al., 2017; Herbelot and Baroni, 2017) or integrate existing morphological generation and inflection tools and regularities (Cotterell et al., 2015; Vulić et al., 2017; Bergmanis et al., 2017) into language models to reduce data sparsity, and improve language modeling for morphologically rich languages. Given the recent success and improved performance with LM-based pre-training methodology (Peters et al., 2018; Howard and Ruder, 2018a) across a wide variety of syntactic and semantic NLP tasks in English, improving language models for other languages might have far-reaching consequences for multilingual NLP in general. Typological information coded in typological databases (Ponti et al., 2018a) offer invaluable support to language modeling (e.g., knowledge on word ordering, morphological regularities). To this end, we now present the results of our enhancement of the CharCNN-LSTM language model that enforces similarity between parameters of morphologically related words, which leads to large perplexity gains across a large number of languages, with the most prominent gains reported for morphologically complex languages.

We now discuss the results of our novel language model with the AP fine-tuning procedure, and its comparison to other language models in our comparison.

Table 5.7 lists all 50 test languages along with their language codes and provides the key statistics of our 50 LM evaluation benchmarks. The statistics include the number of word types in training data, the number of word types occurring in test data but unseen in training, as well as the total number of word tokens in both training and test data, and type-to-token ratios.

Table 5.7 also shows the results for KN5, LSTM-Word, CharCNN-LSTM, and our model with the AP fine-tuning. Furthermore, a visualisation of the CharCNN-LSTM+AP model as a function of type/token ratio is shown in Figure 5.3.

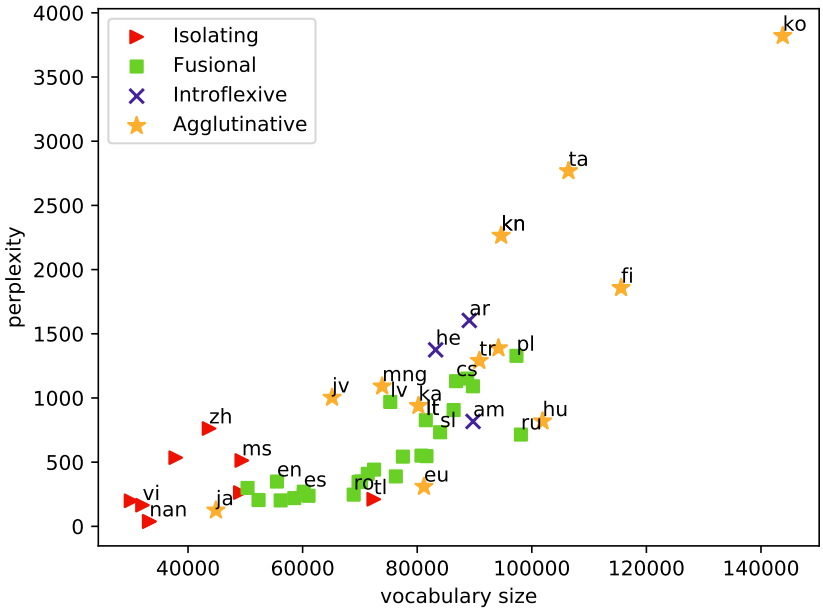



Figure 5.3 Perplexity results with Char-CNN-LSTM+AP (y-axis) in relation to type/token ratio (x-axis). For language codes, see Table 5.7

### 5.10.1 Fine-Tuning the Output Matrix

First, we test the impact of our AP fine-tuning method. As the main finding, the inclusion of fine-tuning into Char-CNN-LSTM (this model is termed +AP) yields improvements on a large number of test languages. The model is better than both strong neural baseline language models for 47/50 languages, and it improves over the original Char-CNN-LSTM LM for 47/50 languages. The largest gains are indicated for the subset of agglutinative MRLs (e.g., 950 perplexity points in Korean, large gains also marked for FI, HE, KA, HU, TA, ET). We also observe large gains for the three introflexive languages included in our study (Amharic, Arabic, Hebrew).

While these large absolute gains may be partially attributed to the exponential nature of the perplexity measure, one cannot ignore the substantial relative gains achieved by our models: e.g., EU ($\Delta PPL$=38) improves more than a fusional language like DA ($\Delta PPL$=24) even with a lower baseline perplexity. This suggests that injecting subword-level information is more straightforward for the former: in agglutinative languages, the mapping between morphemes

and meanings is less ambiguous. Moreover, the number of words that benefit from the injection of character-based information is larger for agglutinative languages, because they also tend to display the highest inflectional synthesis.

For the opposite reasons, we do not surpass Char-CNN-LSTM in a few fusional (IT) and isolating languages (KM, VI). We also observe improvements for Slavic languages with rich morphology (RU, HR, PL). The gains are also achieved for some isolating and fusional languages with smaller vocabularies and a smaller number of rare words, e.g., in Tagalog, English, Catalan, and Swedish. This suggests that our method for fine-tuning the LM prediction is not restricted to MRLs only, and has the ability to improve the estimation for rare words in multiple typologically diverse languages.

### 5.10.2 Language Models, Typological Features, and Corpus Statistics

In the next experiment, we estimate correlation strength of all perplexity scores with a series of independent variables. The variables are 1) type-token ratio in the train data; 2) new word types in the test data; 3) the morphological type of the language among *isolating*, *fusional*, *introflexive*, and *agglutinative*, capturing different aspects related to the morphological richness of a language.

Results with Pearson's $\rho$ (numerical) and $\eta^2$ in one-way ANOVA (categorical) are shown in Table 5.6. Significance tests show p-values $< 1^{-3}$ for all combinations of models and independent variables, demonstrating all of them are good performance predictors. Our main finding indicates that linguistic categories and data statistics both correlate well ($\approx 0.35$ and $\approx 0.82$, respectively) with the performance of language models.

For the categorical variables we compare the mean values per category with the numerical dependent variable. As such, $\eta^2$ can be interpreted as the amount of variation explained by the model - the resulting high correlations suggest that perplexities tend to be homogeneous for languages of the same morphological type, especially so for state-of-the-art models.

This is intuitively evident in Figure 5.3, where perplexity scores of CharCNN-LSTM+AP are plotted against type/token ratio. Isolating languages are placed on the left side of the spectrum as expected, with low type/token ratio and good performance (e.g., VI, ZH). As for fusional languages, sub-groups behave differently. We find that Romance and Germanic languages display roughly the same level of performance as isolating languages, despite their overall larger type/token ratio. Balto-Slavic languages (e.g. CS, LV) instead show both higher perplexities and higher type/token ratio. These differences may be explained in terms of different inflectional synthesis.

Introflexive and agglutinative languages can be found mostly on the right side of the spectrum in terms of performance (see Figure 5.3). Although the languages with highest

absolute perplexity scores are certainly classified as agglutinative (e.g., Dravidian languages such as KN and TA), we also find some outliers in the agglutinative languages (EU) with remarkably low perplexity scores.

### 5.10.3 Corpus Size and Type/Token Ratio

Building on the strong correlation between type/token ratio and model performance from Section 5.10.2, we now further analyse the results in light of corpus size and type/token statistics. The LM datasets for our 50 languages are similar in size to the widely used English PTB dataset (Marcus et al., 1993). As such, we hope that these evaluation datasets can help guide multilingual language modeling research across a wide spectrum of languages.

However, our goal now is to verify that type/token ratio and not absolute corpus size is the deciding factor when unraveling the limitations of standard LM architectures across different languages. To this end, we conduct additional experiments on all languages of the recent Multilingual Wikipedia Corpus (MWC) (Kawakami et al., 2017) for language modeling, using the same setup as before (see Table 5.5). The corpus provides datasets for 7 languages from the same domain as our benchmarks (Wikipedia), and comes in two sizes. We choose the larger corpus variant for each language, which provides about 3-5 times as many tokens as contained in our data sets from Table 5.7.

The results on the MWC evaluation data along with corpus statistics are summarised in Table 5.8. As one important finding, we observe that the gains in perplexity using our fine-tuning AP method extend also to these larger evaluation datasets. In particular, we find improvements of the same magnitude as in the PTB-sized data sets over the strongest baseline model (CharCNN-LSTM) for all MWC languages. For instance, perplexity is reduced from 1781 to 1578 for Russian, and from 365 to 352 for English. We also observe a gain for French and Spanish with perplexity reduced from 282 to 272 and 255 to 243 respectively.

In addition, we test on samples of the Europarl corpus (Koehn, 2005; Tiedemann, 2012) which contains approximately 10 times more tokens than our PTB-sized evaluation data: we use 400K sentences from Europarl for training and testing. However, this data comes from a much narrower domain of parliamentary proceedings: this property yields a very low type/token ratio as visible from Table 5.7. In fact, we find the type/token ratio in this corpus to be on the same level or even smaller than isolating languages (compare with the scores in Table 5.7): 0.02 for Dutch and 0.03 for Czech. This leads to similar perplexities with and without +AP for these two selected test languages. The third EP test language, Finnish, has a slightly higher type/token ratio. Consequently, we do observe an improvement of 10 points in perplexity. A more detailed analysis of this phenomenon follows.

Table 5.9 displays the overall type/token ratio in the training set of these copora. We observe that the MWC has comparable or even higher type/token ratios than the smaller sets despite its increased size. The corpus has been constructed by sampling the data from a variety of different Wikipedia categories (Kawakami et al., 2017): it can therefore be regarded as more diverse and challenging to model. Europarl on the other hand shows substantially lower type/token ratios, presumably due to its narrower domain and more repetitive nature.

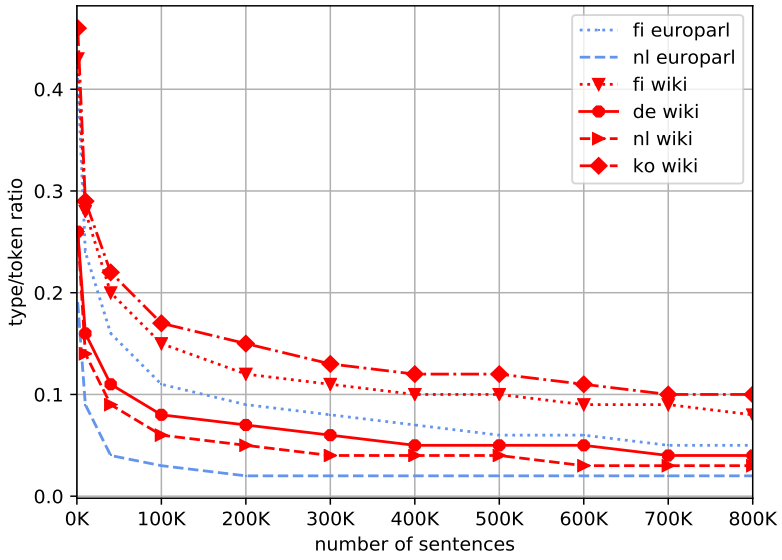



Figure 5.4 Type/token ratio values vs. corpus size. A domain-specific corpus (Europarl) has a lower type/token ratio than a more general corpus (Wikipedia), regardless of the absolute corpus size.

In general, we find that although the type/token ratio decreases with increasing corpus size, the decreasing rate slows down dramatically at a certain point (Herdan, 1960; Heaps, 1978). This depends on the typology of the language and domain of the corpus. Figure 5.4 shows the empirical proof of this intuition. We show the variation of type/token ratios in Wikipedia and Europarl with increasing corpus size. We can see that in a very large corpus of 800K sentences, the type/token ratio in MRLs such as Korean or Finnish stays close to 0.1, a level where we still expect an improvement in perplexity with the proposed AP fine-tuning method applied on top of CharCNN-LSTM.

In order to isolate and verify the effect of the type/token ratio, we now present results on synthetically created data sets where the ratio is controlled explicitly. We experiment with subsets of the German Wikipedia with equal number of sentences (25K)[49], comparable

[49]We split the data into 20K training, 2.5K validation and 2.5K test sentences

number of tokens, but varying type/token ratio. We generate these controlled data sets by clustering sparse bag-of-words sentence vectors with the k-means algorithm, sampling from different clusters, and then selecting the final combinations according to their type/token ratio and the number of tokens. Corpora statistics along with corresponding perplexity scores are shown in Table 5.10, and plotted in Figure 5.5. These results clearly demonstrate and verify that the effectiveness of the AP method increases for corpora with higher type/token ratios. This finding also further supports the usefulness of the proposed method for morphologically-rich languages in particular, where such high type/token ratios are expected.

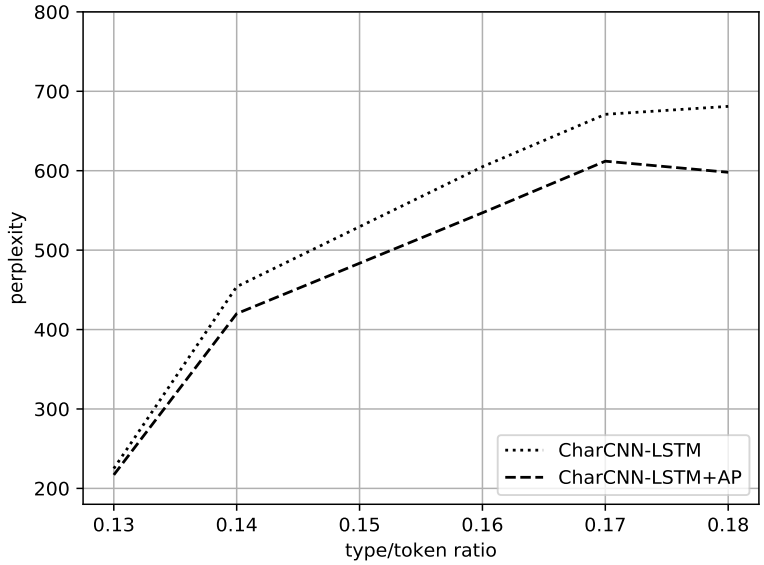



Figure 5.5 Visualisation of results from Table 5.10. The AP method is especially helpful for corpora with high type/token ratios.

## 5.11 Conclusion

In this chapter we have presented a comprehensive language modeling study over a collection of 50 typologically diverse languages. We have demonstrated that typological properties of languages, such as their morphological systems, have an enormous impact on the performance of allegedly "language-agnostic" models.

The languages were carefully selected to represent a wide spectrum of different morphological systems that are found among the world's languages. We have found that the corpus statistics most predictive of LM performance is type-to-token ratio (TTR), as demonstrated by their strong Pearson's correlation. In turn, the value of TTR is motivated by fine-grained typological features that define the type of morphological system within a language. In fact, such features affect the word boundaries and the number of morphemes per word, affecting the word frequency distribution for each language.

We have also observed that injecting character information into word representations is always beneficial because this mitigates the above-mentioned sparsity issues. However, the extent of the gain in perplexity partly depends on some typological properties that regulate the ambiguity of the mapping between morphemes (here modeled as character n-grams) and their meaning. One particular LM challenge is an effective learning of parameters for infrequent words, especially for morphologically-rich languages (MRLs). The methodological contribution of this work is a new neural approach which enriches word vectors at the LM output with subword-level information to capture similar character sequences and consequently to facilitate word-level LM prediction. Our method has been implemented as a fine-tuning step which gradually refines word vectors during the LM training, based on subword-level knowledge extracted in an unsupervised manner from character-aware CNN layers. Our approach yields gains for 47/50 languages in the challenging full-vocabulary setup, with largest gains reported for MRLs such as Korean or Finnish. We have also demonstrated that the gains extend to larger training corpora, and are well correlated with the type-to-token ratio in the training data.

Our study provides new benchmarks and language modeling baselines which should guide the development of next-generation language models focused on the challenging multilingual setting. We hope that NLP/LM practitioners will find the datasets for 50 languages put forth in this work along with benchmarked LMs useful for future developments in (language-agnostic as well as typologically-informed) multilingual language modeling. This study calls for next-generation solutions that will additionally leverage typological knowledge for improved language modeling. Code and data are available at: http://people.ds.cam.ac.uk/dsg40/lmmrl.html.

| Language (code) | Data Stats | | | | | Baseline Models | | | Ours: Fine-Tuning $M^w$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Vocab Size (Train) | New Test Vocab | Number Tokens (Train) | Number Tokens (Test) | Type / Token (Train) | KN5 | LSTM | Char-CNN-LSTM | +AP | Δ +AP |
| × Amharic (am) | 89749 | 4805 | 511K | 39.2K | 0.18 | 1252 | 1535 | 981 | **817** | 164 |
| × Arabic (ar) | 89089 | 5032 | 722K | 54.7K | 0.12 | 2156 | 2587 | 1659 | **1604** | 55 |
| ☐ Bulgarian (bg) | 71360 | 3896 | 670K | 49K | 0.11 | 610 | 651 | 415 | **409** | 6 |
| ☐ Catalan (ca) | 61033 | 2562 | 788K | 59.4K | 0.08 | 358 | 318 | 241 | **238** | 3 |
| ☐ Czech (cs) | 86783 | 4300 | 641K | 49.6K | 0.14 | 1658 | 2200 | 1252 | **1131** | 121 |
| ☐ Danish (da) | 72468 | 3618 | 663K | 50.3K | 0.11 | 668 | 710 | 466 | **442** | 24 |
| ☐ German (de) | 80741 | 4045 | 682K | 51.3K | 0.12 | 930 | 903 | 602 | **551** | 51 |
| ☐ Greek (el) | 76264 | 3767 | 744K | 56.5K | 0.10 | 607 | 538 | 405 | **389** | 16 |
| ☐ English (en) | 55521 | 2480 | 783K | 59.5K | 0.07 | 533 | 494 | 371 | **349** | 22 |
| ☐ Spanish (es) | 60196 | 2721 | 781K | 57.2K | 0.08 | 415 | 366 | 275 | **270** | 5 |
| ★ Estonian (et) | 94184 | 3907 | 556K | 38.6K | 0.17 | 1609 | 2564 | 1478 | **1388** | 90 |
| ★ Basque (eu) | 81177 | 3365 | 647K | 47.3K | 0.13 | 560 | 533 | 347 | **309** | 38 |
| ☐ Farsi (fa) | 52306 | 2041 | 738K | 54.2K | 0.07 | 355 | 263 | 208 | **205** | 3 |
| ★ Finnish (fi) | 115579 | 6489 | 585K | 44.8K | 0.20 | 2611 | 4263 | 2236 | **1858** | 378 |
| ☐ French (fr) | 58539 | 2575 | 769K | 57.1K | 0.08 | 350 | 294 | 231 | **220** | 11 |
| × Hebrew (he) | 83217 | 3862 | 717K | 54.6K | 0.12 | 1797 | 2189 | 1519 | **1375** | 144 |
| ☐ Hindi (hi) | 50384 | 2629 | 666K | 49.1K | 0.08 | 473 | 426 | 326 | **299** | 27 |
| ☐ Croatian (hr) | 86357 | 4371 | 620K | 48.1K | 0.14 | 1294 | 1665 | 1014 | **906** | 108 |
| ★ Hungarian (hu) | 101874 | 5015 | 672K | 48.7K | 0.15 | 1151 | 1595 | 929 | **819** | 110 |
| ▷ Indonesian (id) | 49125 | 2235 | 702K | 52.2K | 0.07 | 454 | 359 | 286 | **263** | 23 |
| ☐ Italian (it) | 70194 | 2923 | 787K | 59.3K | 0.09 | 567 | 493 | **349** | 350 | -1 |
| ★ Japanese (ja) | 44863 | 1768 | 729K | 54.6K | 0.06 | 169 | 156 | 136 | **125** | 11 |
| ★ Javanese (jv) | 65141 | 4292 | 622K | 52K | 0.10 | 1387 | 1443 | 1158 | **1003** | 155 |
| ★ Georgian (ka) | 80211 | 3738 | 580K | 41.1K | 0.14 | 1370 | 1827 | 1097 | **939** | 158 |
| ▷ Khmer (km) | 37851 | 1303 | 579K | 37.4K | 0.07 | 586 | 637 | **522** | 535 | -13 |
| ★ Kannada (kn) | 94660 | 4604 | 434K | 29.4K | 0.22 | 2315 | 5310 | 2558 | **2265** | 293 |
| ★ Korean (ko) | 143794 | 8275 | 648K | 50.6K | 0.22 | 5146 | 10063 | 4778 | **3821** | 957 |
| ☐ Lithuanian (lt) | 81501 | 3791 | 554K | 41.7K | 0.15 | 1155 | 1415 | 854 | **827** | 27 |
| ☐ Latvian (lv) | 75294 | 4564 | 587K | 45K | 0.13 | 1452 | 1967 | 1129 | **969** | 160 |
| ▷ Malay (ms) | 49385 | 2824 | 702K | 54.1K | 0.07 | 776 | 725 | 525 | **513** | 12 |
| ★ Mongolian (mng) | 73884 | 4171 | 629K | 50K | 0.12 | 1392 | 1716 | 1165 | **1091** | 74 |
| ▷ Burmese (my) | 20574 | 755 | 576K | 46.1K | 0.04 | 209 | 212 | 182 | **180** | 2 |
| ▷ Min-Nan (nan) | 33238 | 1404 | 1.2M | 65.6K | 0.03 | 61 | 43 | 39 | 38 | 1 |
| ☐ Dutch (nl) | 60206 | 2626 | 708K | 53.8K | 0.08 | 397 | 340 | 267 | **248** | 19 |
| ☐ Norwegian (no) | 69761 | 3352 | 674K | 47.8K | 0.10 | 534 | 513 | 379 | **346** | 33 |
| ☐ Polish (pl) | 97325 | 4526 | 634K | 47.7K | 0.15 | 1741 | 2641 | 1491 | **1328** | 163 |
| ☐ Portuguese (pt) | 56167 | 2394 | 780K | 59.3K | 0.07 | 342 | 272 | 214 | **202** | 12 |
| ☐ Romanian (ro) | 68913 | 3079 | 743K | 52.5K | 0.09 | 384 | 359 | 256 | **247** | 9 |
| ☐ Russian (ru) | 98097 | 3987 | 666K | 48.4K | 0.15 | 1128 | 1309 | 812 | **715** | 97 |
| ☐ Slovak (sk) | 88726 | 4521 | 618K | 45K | 0.14 | 1560 | 2062 | 1275 | **1151** | 124 |
| ☐ Slovene (sl) | 83997 | 4343 | 659K | 49.2K | 0.13 | 1114 | 1308 | 776 | **733** | 43 |
| ☐ Serbian (sr) | 81617 | 3641 | 628K | 46.7K | 0.13 | 790 | 961 | 582 | **547** | 35 |
| ☐ Swedish (sv) | 77499 | 4109 | 688K | 50.4K | 0.11 | 843 | 832 | 583 | **543** | 40 |
| ★ Tamil (ta) | 106403 | 6017 | 507K | 39.6K | 0.21 | 3342 | 6234 | 3496 | **2768** | 728 |
| ▷ Thai (th) | 30056 | 1300 | 628K | 49K | 0.05 | 233 | 241 | 206 | **199** | 7 |
| ▷ Tagalog (tl) | 72416 | 3791 | 972K | 66.3K | 0.07 | 379 | 298 | 219 | **211** | 8 |
| ★ Turkish (tr) | 90840 | 4608 | 627K | 45K | 0.14 | 1724 | 2267 | 1350 | **1290** | 60 |
| ☐ Ukranian (uk) | 89724 | 4983 | 635K | 47K | 0.14 | 1639 | 1893 | 1283 | **1091** | 192 |
| ▷ Vietnamese (vi) | 32055 | 1160 | 754K | 61.9K | 0.04 | 197 | 190 | **158** | 165 | -7 |
| ▷ Chinese (zh) | 43672 | 1653 | 746K | 56.8K | 0.06 | 1064 | 826 | 797 | **762** | 35 |
| ▷ **Isolating** (avg) | 40930 | 1825 | 759K | 54K | 0.05 | 440 | 392 | 326 | **318** | 8 |
| ☐ **Fusional** (avg) | 73499 | 3532 | 689K | 51.3K | 0.11 | 842 | 969 | 618 | **566** | 52 |
| × **Introflexive** (avg) | 87352 | 4566 | 650K | 49.5K | 0.14 | 1735 | 2104 | 1386 | **1265** | 121 |
| ★ **Agglutinative** (avg) | 91051 | 4687 | 603K | 45K | 0.16 | 1898 | 3164 | 1727 | **1473** | 254 |

Table 5.7 Test perplexities for 50 languages (ISO 639-1 codes sorted alphabetically) in the full-vocabulary prediction LM setup; **Left**: Basic statistics of our evaluation data. **Middle**: Results with the *Baseline LMs*. Note that the absolute scores in the KN5 column are not comparable to the scores obtained with neural models (see Section 5.9). **Right**: Results with Char-CNN-LSTM and our AP fine-tuning strategy. Δ is indicating the difference in performance over the original Char-CNN-LSTM model. The best scoring neural baseline is underlined. The overall best performing neural model for each language is in bold.

| Lang | Corpus | # Vocab | | # Tokens | | Type/Token | Char-CNN-LSTM | +AP |
|---|---|---|---|---|---|---|---|---|
| | | train | test | train | test | train | | |
| nl | EP | 197K | 200K | 10M | 255K | 0.02 | **62** | 63 |
| cs | EP | 265K | 268K | 7.9M | 193K | 0.03 | **180** | 186 |
| en | MWC | 310K | 330K | 5.0M | 0.5M | 0.06 | 365 | **352** |
| es | MWC | 258K | 277K | 3.7M | 0.4M | 0.07 | 255 | **243** |
| fr | MWC | 260K | 278K | 4.0M | 0.5M | 0.07 | 282 | **272** |
| fi | EP | 459K | 465K | 6.8M | 163K | 0.07 | 515 | **505** |
| de | MWC | 394K | 420K | 3.8M | 0.3M | 0.10 | 710 | **665** |
| ru | MWC | 372K | 399K | 2.5M | 0.3M | 0.15 | 1781 | **1578** |
| cs | MWC | 241K | 258K | 1.5M | 0.2M | 0.16 | 2396 | **2159** |
| fi | MWC | 320K | 343K | 1.5M | 0.1M | 0.21 | 5300 | **4911** |

Table 5.8 Results on the larger MWC data set (Kawakami et al., 2017) and on a subset of the Europarl (EP) corpus. Improvements with +AP are not dependent on corpus size, but rather they strongly correlate with the type/token ratio of the corpus.

| | **Type/Token Ratio** | | |
|---|---|---|---|
| **Language** | Our Data | MWC | Europarl |
| Czech | 0.13 | 0.16 | 0.03 |
| German | 0.12 | 0.10 | - |
| English | 0.06 | 0.06 | - |
| Spanish | 0.07 | 0.07 | - |
| Finnish | 0.20 | 0.21 | 0.07 |
| French | 0.07 | 0.07 | - |
| Russian | 0.14 | 0.15 | - |
| Dutch | 0.09 | - | 0.02 |

Table 5.9 Comparison of type/token ratios in the corpora used for evaluation. The ratio is not dependent only on the corpus size but also on the language and domain of the corpus.

| Clusters | # Vocab | | # Tokens | | Type/Token | Char-CNN-LSTM | +AP |
|---|---|---|---|---|---|---|---|
| | train | test | train | test | train | | |
| 2 | 48K | 52K | 382K | 47K | 0.13 | 225 | **217** |
| 2,4 | 69K | 75K | 495K | 62K | 0.14 | 454 | **420** |
| 2,4,5,9 | 78K | 84K | 494K | 62K | 0.16 | 605 | **547** |
| 5,9 | 84K | 91K | 492K | 62K | 0.17 | 671 | **612** |
| 5 | 66K | 72K | 372K | 46K | 0.18 | 681 | **598** |

Table 5.10 Results on German with data sets of comparable size and increasing type/token ratio.

# Part IV

# Function-specific Word Representations

# Motivation

In previous chapters we have seen that it is possible to learn relation-specific vector spaces, such as for semantic similarity (Chapter 3) or entailment (Chapter 4). Furthermore, tying word representations together based on their morphological relations can be helpful especially in a low-data setting (Part III). While the above-mentioned cover several essential aspects, there is a much wider range of semantic and syntactic relations considered important to human language understanding (Binder et al., 2009; Bornkessel-Schlesewsky and Schlesewsky, 2016; Davis, 2016, inter alia).

For instance, many real-world applications will require factual or common-sense knowledge. Consider a restaurant QA system that should be able to answer questions such as *Is schnitzel a German dish?*. One straightforward way of ordering a vector space constructed specifically for this purpose can be to place the vector *German* closest to food items associated with it, such as *schnitzel* or *sauerkraut*. Note that there are many different aspects to similarity, and the function-specific relation between *German* and *schnitzel* is reflecting merely their similarity in terms of food. This is different to for example the *function-specific* relation between *German* - *Hessen* (federal states), *German* - *Volkswagen* (companies), or *German* - *France* (countries). If the vector space is trained specifically for the function-specific food relation, vector positions in the space will not conflate different semantic relations. Having a neural model reply with which food items are a typical German dish then becomes a trivial task: We can simply retrieve the food item vectors (*schnitzel*, *sauerkraut*) closest to the question word (*German*).

Likewise, the idea of such *relation-specific* or *function-specific* vector spaces can be applied to other tasks. For example, cognitive science has long studied the likely distinct processing of objects (typically nouns) and actions (typically verbs) in human language understanding (Vigliocco et al., 2011), and there is a range of well-defined NLP evaluations covering this type of information (McRae et al. (1997), McRae et al. (1998), Coecke et al. (2010), Grefenstette and Sadrzadeh (2011a), Kartsaklis and Sadrzadeh (2014)). If a model should be able to retrieve objects commonly co-occurring with verbs (*thematic-fit* or *selectional preference*), we can construct a vector space such that the verbs will be closest to their objects.

A function-specific approach to modeling thereby allows us to generalise beyond the classic semantic relations between words such as described in WordNet (Beckwith et al., 1991), and also learn representations for multi-variable semantic relations between different semantic or syntactic groups of words. In this chapter we introduce a modeling approach for learning such custom, *function-specific* vector spaces. We choose to demonstrate its applicability on the example of the *subject-verb-object* structure (as illustrated in Figure

5.6) since it is covered by established NLP tasks (*event similarity*, *thematic-fit* or *selectional preference*), and hope the model can find application for a range of relation-specific or function-specific phenomena (Future Work).

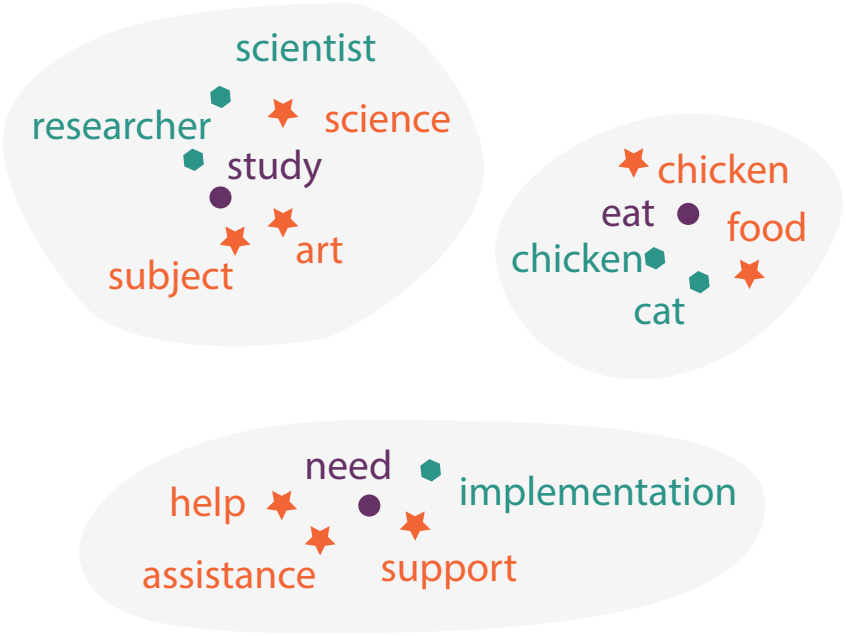



Figure 5.6 Illustration of three neighbourhoods in a function-specific space trained for the SVO structure. The space is structured by type (i.e. S, V, and O) and optimised such that vectors for plausible SVO combinations will be close. Note that one word can have several vectors, for example a *chicken* can either be a subject or an object. See table 6.1 for more examples extracted from the trained model.

This chapter is based on the following publication:

- **Daniela Gerz**, Ivan Vulić, Marek Rei, Roi Reichart, Anna Korhonen. "Multidirectional Associative Optimization of Function-Specific Word Representations." (*Under Review*)

# Chapter 6

# Function-specific Word Representations

We present a neural framework for learning associations between interrelated groups of words such as the ones found in Subject-Verb-Object (SVO) structures. Our model induces a joint function-specific word vector space, where vectors of e.g. plausible SVO compositions lie close together. The model retains information about word group membership even in the joint space, and can thereby effectively be applied to a number of tasks reasoning over the SVO structure. We show the robustness and versatility of the proposed framework by reporting state-of-the-art results on the tasks of estimating selectional preference and event similarity. The results indicate that the combinations of representations learned with our task-independent model outperform task-specific architectures from prior work, while reducing the number of parameters by up to 95%.

## 6.1 Introduction

Word representations are in ubiquitous usage across all areas of natural language processing (NLP) (Collobert et al., 2011; Chen and Manning, 2014; Melamud et al., 2016). Standard approaches rely on the distributional hypothesis (Harris, 1954; Schütze, 1993) and learn a *single* word vector space based on word co-occurrences in large text corpora (Mikolov et al., 2013b; Pennington et al., 2014; Bojanowski et al., 2017). This purely context-based training produces general word representations that capture the broad notion of semantic relatedness and conflate a variety of possible semantic relations into a single space (Hill et al., 2015; Schwartz et al., 2015). However, this mono-faceted view of meaning is a well-known deficiency in NLP applications (Faruqui, 2016; Mrkšić et al., 2017) as it fails to distinguish between fine-grained word associations.

In this work we propose to learn a joint *function-specific* word vector space that accounts for the different roles and functions a word can take in text. The space can be trained for

a specific structure, such as SVO, and each word in a particular role will have a separate representation. Vectors for plausible SVO compositions will then be optimized to lie close together, as illustrated by Figure 5.6. For example, the verb vector *study* will be close to plausible subject vectors *researcher* or *scientist* and object vectors *subject* or *art*. For words that can occur as either subject or object, such as *chicken*, we obtain separate vectors for each role: one for *chicken* as *subject* and another for *chicken* as *object*. The resulting representations capture more detailed associations in addition to basic distributional similarity and can be used to construct representations for the whole SVO structure.

To validate the effectiveness of our representation framework in language applications, we focus on modeling a prominent linguistic phenomenon: a general model of *who does what to whom* (Gell-Mann and Ruhlen, 2011). In language, this event understanding information is typically captured by the SVO structures and, according to the cognitive science literature, is well aligned with how humans process sentences (McRae et al., 1997, 1998; Grefenstette and Sadrzadeh, 2011a; Kartsaklis and Sadrzadeh, 2014); it reflects the likely distinct storage and processing of objects (typically nouns) and actions (typically verbs) in the brain (Caramazza and Hillis, 1991; Damasio and Tranel, 1993).

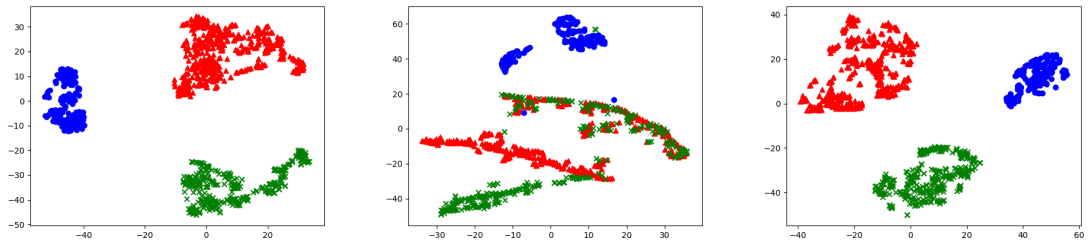
| **Word** | **Nearest Neighbours** |
|---|---|
| **Subject** | |
| memory | dream, feeling, shadow, sense, moment, consciousness |
| country | state, nation, britain, china, uk, europe, government |
| student | pupil, participant, learner, candidate, trainee, child |
| **Verb** | |
| see | saw, view, expect, watch, notice, witness |
| eat | drink, consume, smoke, lick, swallow, cook, ingest |
| avoid | eliminate, minimise, anticipate, overcome, escape |
| **Object** | |
| virus | bacteria, infection, disease, worm, mutation, antibody, bug |
| beer | ale, drink, pint, coffee, tea, wine, soup, champagne |
| **Joint SVO** | |
| study (V) | researcher (S), scientist (S), subject (O), art (O), science (O) |
| eat (V) | food (O), cat (S), dog (S) |
| need (V) | help (O), implementation (S), support (O), assistance (O) |

Table 6.1 Nearest neighbours of selected words in function-specific word vector spaces.

The quantitative results are reported on two established test sets for compositional event similarity (Grefenstette and Sadrzadeh, 2011a; Kartsaklis and Sadrzadeh, 2014). This task requires reasoning over SVO structures and quantifies the plausibility of the SVO combinations by scoring them against human judgments. We report consistent gains over established word representation methods, as well as over two recent tensor-based architectures (Tilk et al., 2016; Weber et al., 2018) which are designed specifically for solving the event similarity task.

Furthermore, we investigate the generality of our approach by also applying it to other types of structures. We conduct additional experiments in a 4-role setting, where indirect objects are also modeled, along with a *selectional preference* evaluation of 2-role SV and VO relationships (Chambers and Jurafsky, 2010; Van de Cruys, 2014), yielding the highest scores on several established benchmarks.

## 6.2 Function-specific Representation Space



(a) Predicting $n \rightarrow 1$ (b) Predicting $1 \rightarrow n$ (c) Our multidirectional approach

Figure 6.1 The directionality of prediction in neural models is important. Representations can be of varying quality depending on whether they are induced at the input or output side of the model. Our multidirectional approach resolves this problem by training on shared representations in all directions.

**Objectives.** We propose to induce function-specific vector spaces which enable a better model of associations between concepts and consequently improved event representations by encoding the relevant information directly into the parameters for each word during training. Word vectors offer several advantages over tensors: a large reduction in parameters and fixed dimensionality across concepts. This facilitates their reuse and transfer across different tasks. For this reason, we find our multidirectional training to deliver good performance: the same function-specific vector space achieves state-of-the-art scores across multiple related tasks, previously held by task-specific models.

We require a flexible model that can **a)** represent words in a distributed and interconnected manner, such that **b)** the model has a high capacity for learning associations between all words and their representations. Formally, our goal is to model the mutual associations (co-occurrences) between $N$ variables, where the vocabularies of each variable can partially overlap. We induce an embedding or a look-up matrix (i.e., a vector space), $\mathbb{R}^{|V_i| \times d}$ for each variable $i = 1, \ldots, N$, where $|V_i|$ corresponds to the vocabulary size of the $i$-th variable. For consistency, the vector dimensionality $d$ is kept equal across all variables.

**Multiple Variables.** Without loss of generality we present a model which learns $N = 3$ function-specific embedding spaces: we generically refer to those as $A$, $B$, and $C$. Note that the model is not limited to this setup, as we show later in Section 6.5.1. We refer to each embedding space as *variable*, and to a single instance from the space as *vector* or *embedding*. $A$, $B$ and $C$ might be interrelated phenomena, and we aim for a model which can reliably score the plausibility of combining three vectors taken from the vector spaces.[50] In addition to the full joint prediction, we aim for any two vector combinations ($\vec{A}\vec{B}$, $\vec{B}\vec{C}$, $\vec{C}\vec{A}$) to have plausible scores of their own. Observing relations between words inside their respective embedding space (i.e., $A$, $B$, $C$ separately) is another desirable feature.

**Naïve Solution.** We first introduce an obvious but inefficient solution which motivates our approach. The variables $\{A, B, C\}$ can be split into two-variable subsets. This leads to three possible two-variable combinations: $AB^T$, $BC^T$ and $CA^T$. By implementing the factorisation with a neural model such as SGNS, the number of combinations is doubled since neural models are *directed* by default based on the direction of the prediction ($AB^T + b_0$, $BA^T + b_1$, $BC^T + b_2$, ...). We thus have six different networks in practice. Each network is instantiated as a one-layer feedforward network without any non-linearity, and with a network-specific bias vector $b_\mu$ to model each of the six directions. The networks are trained based on cross-entropy loss.

While training six networks in this *asynchronous* manner covers all possible directions, it is suboptimal: **1)** the information cannot flow between variables, which makes a reliable joint prediction impossible; **2)** all parameters are network-specific and trained from scratch for each network: it means that the model learns overlapping information several times stored in different parameters, making it large and computationally inefficient.

**Directionality.** To further highlight that the asynchronous model is suboptimal, we illustrate the effect of *prediction directionality* on the induced representations relying on a simple example. In languages that exhibit noun gender (e.g., German has three noun genders), knowing the correct gender is crucial for the correct inflection. Imagine two spaces representing words

[50] As mentioned in the introduction, the three variables for which embedding spaces are induced can be Subject, Verb, and Object forming the SVO structure.

($A_{word}$) and the three genders ($B_{gender}$), respectively. As most nouns are assigned exactly one gender, this problem is an *n:1* assignment case. Consequently, we expect a word vector space customised for this purpose to show three clearly separated clusters. Figure 6.1 visualises obtained representations.[51] Figure 6.1a plots the vector spaces when we use words on the input side of the model and predict their gender: $A_{word} \rightarrow B_{gender}$, *n:1* assignment. In the opposite direction ($B_{gender} \rightarrow A_{word}$, *1:n* assignment) we do not observe the same trends (Figure 6.1b).

Representations for other and more complex phenomena suffer from the same issue. For example, the verb *eat* can take many arguments corresponding to various food items such as *pizza*, *beans*, or *kimchi*. A more specific verb such as *embark* might take only a few arguments such as *journey*, whereas *journey* might be fairly general and can co-occur with many other verbs themselves. We thus effectively deal with an *n:m* assignment case, which might be inclined towards *1:n* or *n:1* entirely depending on the words in question. Therefore, it is unclear whether one should rather construct a model predicting *verb* $\rightarrow$ *object* or *object* $\rightarrow$ *verb*. We resolve this fundamental design question by training representations in a *multidirectional* way with a *joint loss* function. Figure 6.1c shows how this method learns accurately clustered representations without having to make directionality assumptions.

## 6.3 Multidirectional Synchronous Learning

The multidirectional neural representation learning model takes a list of $N$ groups of words $(G_1, \ldots, G_N)$, factorises it into all possible "group-to-group" sub-models, and trains them jointly by combining objectives based on skip-gram negative sampling Mikolov et al. (2013a,b). We learn a joint function-specific word vector space by using sub-networks that each consume one group $G_i$ on the input side and predict words from a second group $G_j$ on the output side, $i, j = 1, \ldots, N; i \neq j$. All sub-network losses are tied into a single joint loss and all groups $G_1, \ldots, G_n$ are shared between the sub-networks.

**Sub-Network Architecture.** We first factorise groups into sub-networks, representing all possible directions of prediction. Two groups would lead to two sub-networks $A \rightarrow B$ and $B \rightarrow A$; three groups lead to six sub-networks.

Similar to Mikolov et al. (2013a,b), we calculate the dot-product between two word vectors to quantify their association. For instance, the sub-network $A \rightarrow B$ computes its

[51] We train on 10K randomly selected German nouns from a German-English dictionary obtained from dict.cc, and train a 25-dim model for 24 epochs. Points in the figures show 1K words randomly selected from the 10K trained vocabulary. The embedding spaces have been mapped to 2D with tSNE.

prediction:

$$P_{A\rightarrow B} = \sigma(\vec{a} \cdot B_e^T + \vec{b}_{ab}) \tag{6.1}$$

where $\vec{a}$ is a word vector from the input group $A$, $B_e$ is the word embedding matrix for the target group $B$, $\vec{b}_{ab}$ is a bias vector, and $\sigma$ is the sigmoid function. The loss of each sub-network is computed using cross-entropy between this prediction and the correct labels:

$$\mathscr{L}_{A\rightarrow B} = cross\_entropy(P_{A\rightarrow B}, L_{A\rightarrow B}) \tag{6.2}$$

where $L_{A\rightarrow B}$ are one-hot vectors corresponding to the correct predictions. We leave experiments with more sophisticated sub-network designs for future work.

**Synchronous Joint Training.** We integrate all sub-networks into one joint model via two following mechanisms:

**(1) Shared Parameters.** The three embedding matrices referring to groups $A$, $B$ and $C$ are shared across all sub-networks. That is, we train one matrix per group, regardless of whether it is being employed at the input or the output side of any sub-network. This leads to a substantial reduction in the model size. For example, with a vocabulary of $50,000$ words and 25-dimensional vectors we work only with 1.35M parameters. Comparable models for the same tasks are trained with much larger sets of parameters: 26M or even up to 179M when not factorised (Tilk et al., 2016). Our modeling approach thus can achieve more that 95% reduction in the number of parameters.

**(2) Joint Loss.** We also train all sub-networks with a single joint loss and a single backward pass. We refer to this manner of joining the losses as *synchronous*: it synchronises the backward pass of all sub-networks. This could also be seen as a form of multi-task learning, where each sub-network optimises the shared parameters for a different task (**?**). In practice, we perform a forward pass in each direction separately, then join all sub-network cross-entropy losses and backpropagate this joint loss through all sub-networks in order to update the parameters. The different losses are combined using addition:

$$\mathscr{L} = \sum_{\mu} \mathscr{L}_{\mu} \tag{6.3}$$

where $\mu$ iterates over all the possible sub-networks, $\mathscr{L}_{\mu}$ is the corresponding loss from one network, and $\mathscr{L}$ the overall joint loss.

When focusing on the SVO structures, the model will learn one joint space for the three groups of embeddings (one for $S$, $V$ and $O$). The 6 sub-networks all share parameters and optimization is performed using the joint loss:

$$\begin{aligned}\mathscr{L} = &\mathscr{L}_{S\rightarrow V} + \mathscr{L}_{V\rightarrow S} + \mathscr{L}_{V\rightarrow O} \\ &+ \mathscr{L}_{O\rightarrow V} + \mathscr{L}_{S\rightarrow O} + \mathscr{L}_{O\rightarrow S}\end{aligned} \tag{6.4}$$

The vectors from the induced function-specific space can then be composed by standard composition functions (Milajevs et al., 2014) to yield *event representations* (Weber et al., 2018), that is, representations for the full SVO structure.

## 6.4 Evaluation Setup

**Preliminary Task: Pseudo-Disambiguation.** In the first evaluation, we adopt a standard *pseudo-disambiguation* task from the selectional preference literature (Rooth et al., 1999; Bergsma et al., 2008; Erk et al., 2010; Chambers and Jurafsky, 2010; Van de Cruys, 2014). For the three-variable (S-V-O) case, the task is to score a *true* triplet (i.e., the (S-V-O) structure attested in the corpus) above all *corrupted* triplets (S-V'-O), (S'-V-O), (S-V-O'), where S', V' and O' denote subjects and objects randomly drawn from their respective vocabularies. Similarly, for the two-variable setting, the task is to express a higher preference towards the attested pairs (V-O) or (S-V) over corrupted pairs (V-O') or (S'-V). We report accuracy scores, i.e., we count all items where *score(true)* $>$ *score(corrupted)*.

This simple pseudo-disambiguation task serves as a preliminary sanity check: it can be easily applied to a variety of training conditions with different variables. However, as pointed out by Chambers and Jurafsky (2010), the performance on this task is strongly influenced by a number of factors such as vocabulary size and the procedure for constructing corrupted examples. Therefore, we additionally evaluate our models on a number of other established datasets (Sayeed et al., 2016).

**Event Similarity (3 Variables: SVO).** A standard task to measure the plausibility of 3-variable SVO structures (i.e., *events*) is *event similarity* (Grefenstette and Sadrzadeh, 2011a; Weber et al., 2018): the goal is to score similarity between SVO triplet pairs and correlate the similarity scores to human-elicited similarity judgements. Robust and flexible event representations are important to many core areas in language understanding such as script learning, narrative generation, and discourse understanding (Chambers and Jurafsky, 2009; Pichotta and Mooney, 2016; Modi, 2016; Weber et al., 2018). We evaluate event similarity on two benchmarking data sets: **GS199** (Grefenstette and Sadrzadeh, 2011a) and **KS108** (Kartsaklis and Sadrzadeh, 2014). GS199 contains 199 pairs of *SVO* triplets/events. In the GS199 data set only the $V$ is varied, while $S$ and $O$ are fixed in the pair: this evaluation

prevents the model from relying only on simple lexical overlap for similarity computation.[52] KS108 contains 108 event pairs for the same task, but is specifically constructed without any lexical overlap between the events in each pair.

For this task our specialised representations are composed into a single *event representation/vector*. Following prior work, we compare cosine similarity of event vectors to averaged human scores and report Spearman's $\rho$ correlation with human scores. We compose function-specific vectors into event vectors using simple addition and multiplication, as well as more sophisticared compositions from prior work (Milajevs et al., 2014, *inter alia*). The summary is provided in Table 6.2.

| Composition | Reference | Formula |
|---|---|---|
| Verb only | Milajevs et al. (2014) | $\vec{V}$ |
| Addition | Mitchell and Lapata (2008) | $\vec{S}+\vec{V}+\vec{O}$ |
| Copy Object | Kartsaklis et al. (2012) | $\vec{S}\odot(\vec{V}\times\vec{O})$ |
| Concat | Edelstein and Reichart (2016) | $[\vec{S},\vec{V},\vec{O}]$ |
| Concat Addition | Edelstein and Reichart (2016) | $[\vec{S},\vec{V}]+[\vec{V},\vec{O}]$ |
| Network | Ours | $\vec{S}\vec{V}^T+\vec{V}\vec{O}^T+\vec{S}\vec{O}^T$ |

Table 6.2 Composition functions used to obtain event vectors from function-specific vector spaces. $+$: addition, $\odot$: element-wise multiplication, $\times$: dot product. $[\cdot,\cdot]$: concatenation.

**Thematic-Fit Evaluation (2 Variables: SV and VO).** Similarly to the 3-variable setup, we also evaluate the plausibility of *SV* and *VO* pairs separately in the 2-variable setup. The thematic-fit evaluation (Sayeed et al., 2016) quantifies the extent to which a noun fulfils the selectional preference of a verb given a role (i.e., agent:S, or patient:O) (McRae et al., 1997). We evaluate our 2-variable function-specific spaces on two standard benchmarks: **1) MST1444** (McRae et al., 1998) contains 1,444 word pairs where humans provided thematic fit ratings on a scale from 1 to 7 for each noun to score the plausibility of the noun taking the agent role, and also taking the patient role.[53] **2) PADO414** (Padó, 2007) is similar to MST1444, containing 414 pairs with human thematic fit ratings, where role-filling nouns were selected to reflect a wide distribution of scores for each verb. We compute plausibility by simply taking the cosine similarity between the verb vector (from the $V$ space) and the noun vector from the appropriate function-specific space ($S$ space for agents; $O$ space for patients). We again report Spearman's $\rho$ correlation scores.

[52] For instance, the phrases *'people run company'* and *'people operate company'* have a high similarity score of 6.53, whereas *'river meet sea'* and *'river satisfy sea'* have been given a low score of 1.84.

[53] Using an example from Sayeed et al. (2016), the human participants were asked "how common is it for a {snake, monster, baby, cat} to frighten someone/something" (agent role) as opposed to "how common is it for a {snake, monster, baby, cat} to be frightened by someone/something" (patient role).

## 6.5 Experiments and Results

**Training Data.** We parse the ukWaC corpus (Baroni et al., 2009) and the British National Corpus (BNC) (Leech, 1992) using the Stanford Parser with Universal Dependencies v1.4 (Chen and Manning, 2014; Nivre et al., 2016) and extract co-occurring subjects, verbs and objects. All words are lowercased and lemmatised, and tuples containing non-alphanumeric characters are excluded. We also remove tuples with (highly-frequent) pronouns as subjects, and filter out training examples containing words with frequency lower than 50. After preprocessing, the final training corpus comprises 22M SVO triplets in total. Table 6.3 additionally shows training data statistics when training in the 2-variable setup (SV and VO) and in the 4-variable setup (when adding indirect objects: SVO+iO). We report the number of examples in training and test sets, as well as vocabulary sizes and most frequent words across different categories.

**Hyperparameters.** We train with batch size 128, and use Adam for optimisation (Kingma and Ba, 2015) with a learning rate 0.001. All gradients are clipped to a maximum norm of 5.0. All models were trained with the same fixed random seed. We train 25-dimensional vectors for all setups (2/3/4 variables), and we additionally train 100-dimensional vectors for the 3-variable (SVO) setup.

| **Data set** | **Train** | **Test** |
|---|---|---|
| **SVO+iO** | 187K | 15K |
| **SVO** | 22M | 214K |
| | **Vocab size** | **Most frequent words** |
| S | 22K | people, one, company, student, government, group |
| V | 5K | have, take, include, provide, make, give, offer, use |
| O | 15K | place, information, way, number, opportunity, time |
| **SV** | 69M | 232K |
| | **Vocab size** | **Most frequent words** |
| S | 45K | people, what, one, these, company, thing, student |
| V | 19K | be, have, say, take, go, make, include, come, provide |
| **VO** | 84M | 240K |
| | **Vocab size** | **Most frequent words** |
| V | 9K | have, take, use, make, provide, give, get, see |
| O | 32K | information, time, service, way, people, place |

Table 6.3 Training data statistics.

### 6.5.1 Results and Analysis

**Pseudo-Disambiguation.** Accuracy scores on the pseudo-disambiguation task in the 2/3/4-variable setups are summarised in Table 6.4.[54] We find consistently high pseudo-disambiguation scores (>0.94) across all setups. As mentioned in Section 6.4, this initial evaluation already suggests that our model is able to capture associations between interrelated variables which are instrumental to modeling SVO structures and, more generally, constructing multi-variable event representations.

| Model | Accuracy |
|---|---|
| **4 Variables** | |
| SVO+iO | **0.950** |
| **3 Variables: SVO** | |
| Van de Cruys (2009) | 0.874 |
| Van de Cruys (2014) | 0.889 |
| Tilk et al. (2016) (our reimplementation) | 0.937 |
| Ours | **0.943** |
| **2 Variables** | |
| Rooth et al. (1999) | 0.720 |
| Erk et al. (2010) | 0.887 |
| Van de Cruys (2014) | 0.880 |
| Ours: SV | **0.960** |
| Ours: VO | **0.972** |

Table 6.4 Pseudo-disambiguation: accuracy scores.

**Event Similarity.** We now test correlations of SVO-based event representations composed from function-specific vector spaces (see Table 6.2 again) to human scores in the event similarity task. A summary of the main results is provided in Table 6.5. We also report best baseline scores from prior work.

The main finding is that our model based on function-specific word vectors outperforms previous state-of-the-art scores on both datasets. It is crucial to note that different modeling approaches and configurations from prior work held previous peak scores on the two evaluation sets.[55]

Interestingly, by relying only on the isolated verb vectors (i.e., by completely discarding the knowledge stored in $S$ and $O$ vectors), we can already obtain reasonable correlation scores. Given that the model did consume the knowledge on subjects and objects during the

[54] We also provide baseline scores taken from prior work, but the reader should be aware that the scores may not be directly comparable due to the dependence of this evaluation on factors such as vocabulary size and sampling of corrupted examples (Chambers and Jurafsky, 2010; Sayeed et al., 2016).

[55] Note the two tasks are inherently different. KS108 requires similarity between plausible triplets. Using the network score directly (which is a scalar, see Table 6.2) is not suitable for KS108 as all KS108 triplets are plausible and scored highly. This is reflected in the results in Table 6.5.

joint multidirectional training, this is an indicator that the single verb vector space already stores some selectional preference information.

| Model | Reference | Spearman's $\rho$ Correlation GS199 | KS108 |
|---|---|---|---|
| Copy Object W2V | Milajevs et al. (2014) | 0.46 | 0.66 |
| Addition KS14 | Milajevs et al. (2014) | 0.28 | 0.73 |
| | Tilk et al. (2016) | 0.34 | - |
| | Weber et al. (2018) | - | 0.71 |
| **Ours: SVO d100** | | | |
| Verb only | Ours | 0.34 | 0.63 |
| Addition | Ours | 0.27 | **0.76** |
| Concat | Ours | 0.26 | **0.75** |
| Concat Addition | Ours | 0.32 | **0.77** |
| Copy Object | Ours | 0.40 | 0.52 |
| Network | Ours | **0.53** | - |

Table 6.5 Results on the event similarity task. Best baseline score is underlined, and the best overall result is provided in **bold**.

**Thematic-Fit Evaluation.** Correlation scores on two thematic-fit evaluation data sets are summarised in Table 6.6. We also report results with representative baseline models for the task: 1) a TypeDM-based model (Baroni and Lenci, 2010), further improved by Greenberg et al. (2015a,b) (**G15**), and 2) current state-of-the-art tensor-based neural model by Tilk et al. (2016) (**TK16**).

| Setup | | Baselines | | Ours | |
|---|---|---|---|---|---|
| Dataset | Eval | G15 | TK16 | SVO (d=100) | SV-VO (d=25) |
| MST1444 | SV | 0.36 | - | **0.37** | 0.31 |
| | VO | 0.34 | - | **0.35** | 0.35 |
| | full | 0.33 | **0.38** | 0.36 | 0.34 |
| PADO414 | SV | 0.54 | - | 0.38 | **0.55** |
| | VO | 0.53 | - | 0.54 | **0.61** |
| | full | 0.53 | 0.52 | 0.45 | **0.58** |

Table 6.6 Results on the 2-variable thematic-fit evaluation. Spearman's $\rho$ correlation scores reported.

We find that vectors taken from the model trained in the joint 3-variable SVO setup perform on a par with state-of-the-art models also in the 2-variable evaluation on SV and VO subsets. Vectors trained explicitly in the 2-variable setup using three times more data lead to substantial improvements on PADO414. As a general finding, our thematic fit method based on function-specific spaces leads to peak performance across on both data sets. The results are similar with 25-dim SVO vectors.

Our model is also more light-weight than the baselines: we do not require a full (tensor-based) neural model, but simply function-specific word vectors to reason over thematic fit.

To further verify the importance of joint multidirectional training, we have also compared our function-specific vectors against standard single-space word vectors (Mikolov et al., 2013b). The results indicate the superiority of function-specific spaces: respective correlation scores on MST1444 and PADO414 are 0.28 and 0.41 (vs 0.34 and 0.58 with our model).

It is interesting to note that we obtain state-of-the-art scores calculating cosine similarity of vectors taken from *two distinct vector spaces*. Despite two distinct matrices, the model in practice learns a joint space where co-occurring words of different categories lie close to each other. This allows us to either look at each space on its own or join them together depending on the application.

**Qualitative Analysis.** We retrieve nearest neighbours from the function-specific ($S$, $V$, $O$) space, shown in Figure 5.6. We find that the nearest neighbours indeed reflect the relations required to model the SVO structure. For instance, the closest subjects/agents to the verb *eat* are *cat* and *dog*. The closest objects to *need* are three plausible nouns: *help*, *support*, and *assistance*. As the model has information about group membership, we can also filter and compare nearest neighbours in single-group subspaces. For example, we find subjects similar to the subject *memory* are *dream* and *feeling*, and objects similar to *beer* are *ale* and *pint*.

| | **async sep** | **async shared** | **sync sep** | **sync shared** |
|---|---|---|---|---|
| **3 Variables** | | | | |
| KS108 Verb only | 0.56 | 0.48 | 0.58 | **0.60** |
| KS108 Addition | 0.51 | 0.66 | 0.73 | **0.78** |
| GS199 Verb only | 0.24 | 0.26 | 0.26 | **0.34** |
| GS199 Network | 0.10 | 0.40 | 0.28 | **0.52** |
| **2 Variables** | | | | |
| MST1444 | 0.17 | 0.10 | 0.30 | **0.39** |
| PADO414 | 0.41 | 0.21 | **0.44** | **0.44** |

Table 6.7 Evaluation of different model variants.

**Model Variants.** We also conduct an ablation study that compares different model variants. The variants are constructed by varying 1) the training regime: asynchronous (*async*) vs synchronous (*sync*)[56] and 2) the type of parameter sharing: training on separate parameters for each sub-network (*sep*)[57] or training on shared variables (*shared*). Table 6.7 shows the results with the model variants, demonstrating that both aspects (i.e., shared parameters and synchronous training) are important to reach improved overall performance. We reach the peak scores on all evaluation sets using the *sync+shared* variant. We suspect that asynchronous training deteriorates performance because each sub-network overwrites the updates

[56]In the asynchronous setup we update the shared parameters per sub-network directly based on their own loss, instead of relying on the joint synchronous loss as in Section **??**.

[57]With separate parameters we merge vectors from "duplicate" vector spaces by non-weighted averaging.

of other sub-networks as their training is not tied through a joint loss function. On the other hand, the synchronous training regime guides the model towards making updates that can benefit all sub-networks.

## 6.6 Conclusion and Future Work

We presented a novel multidirectional neural framework for learning function-specific word representations, which can be easily composed into multi-word representations to reason over event similarity and thematic fit. We induce a joint vector space in which several groups of words (e.g., S, V, and O words forming the SVO structures) are represented while taking into account the mutual associations between the groups. We found that resulting function-specific vectors yield state-of-the-art results on established benchmarks for the tasks of estimating event similarity and evaluating thematic fit, previously held by task-specific methods.

# Part V

# Conclusion

# Conclusion

> If you talk to [someone] in a language [he or she] understands, that goes to [the person's] head. If you talk to [somebody] in [his or her] language, that goes to [the] heart.
>
> *Nelson Mandela*

Representation learning is a key research area within natural language processing as vector representations form one of the most fundamental building blocks of modern machine learning models (Collobert and Weston, 2008; Collobert et al., 2011; Bengio et al., 2003; Mikolov et al., 2010; Sutskever et al., 2014; Cho et al., 2014; Chen and Manning, 2014; Henderson et al., 2017, inter alia). A major focus of prior work has been on training representations based on the distributional hypothesis (Harris, 1954; Firth, 1968) with an objective that predicts words co-occurring in context (Turney et al., 2010; Mikolov et al., 2013b,a; Pennington et al., 2014; Bojanowski et al., 2017) using large corpora (Al-Rfou et al., 2013). This type of training has been found to produce representations working especially well for noun relatedness and similarity (Rubenstein and Goodenough, 1965; Finkelstein et al., 2002; Bruni et al., 2014; Hill et al., 2015). However, there are many semantic relations relevant to human language understanding (Quillian, 1966, 1967; Caramazza and Shelton, 1998; Warrington, 1975; Riddoch et al., 1988; Rice et al., 2015; de Heer et al., 2017, inter alia), and purely distributional approaches tend to conflate them (Mrkšić et al., 2017).

This dissertation aimed to go beyond the limitations of the distributional hypothesis, and provides evaluation sets and modeling approaches primarily targeting previously under-resourced phenomena. Our research objectives were to expand evaluation sets to provide a broader cover and more diverse range of semantic relations and languages, in addition to introducing new modeling approaches to tackle the limitations we found.

In particular, as many prior evaluations have focused on semantic similarity for nouns, we found there is a lack of both wide-coverage evaluation resources and models for other word

types and semantic relations. Part II contributed a novel intrinsic evaluation resource for verb similarity, SimVerb-3500 (Gerz et al., 2016). SimVerb provides human similarity ratings for 3,500 verb pairs, covering normed verb types from the USF free-association database, and providing at least three examples for every VerbNet class. Thanks to its large size, we were able to conduct an analysis spanning selected subsets of the data, and concluded that distributional models trained from raw text perform very poorly on low-frequency and highly polysemous verbs. This analysis strengthens the need for more thorough, wide-coverage evaluation of other semantic relations vital to human language understanding. One such relation is lexical entailment. We introduced a graded evaluation set, HyperLex (Vulić et al., 2017), for lexical entailment. Lexical entailment had previously only been evaluated as binary, despite clear evidence from cognitive science that it is a gradual relation (Hampton, 2007, inter alia). Similar to established evaluation methodology for semantic similarity (Chapter 3), other semantic relations can be evaluated more precisely with graded scores by calculating the correlation of the model output to human judgments.

Another dimension we have considered is the influence of typological factors on learning representations. To date there are no representation evaluation resources featuring comparable human judgments on a very large and highly diverse set of languages. We therefore concentrated on the task of word-level language modeling in Part III as a proxy to analyse the influence of word representations across a set of 50 typologically-diverse languages, and observed that indeed, typological factors can have an enormous impact. Languages were carefully selected to represent a wide spectrum of different morphological systems that are found among the world's languages. We have found that the corpus statistic most predictive of performance is the type-to-token (TTR) ratio (Table 5.9). The TTR results from fine-grained typological features of the language that define word boundaries, thereby affecting the frequency distributions of the language. Especially in morphologically-rich languages we encounter a high number of low frequency words, which are extremely hard to model with purely neural approaches (Shareghi et al., 2019), and correlate highly with low model performance (Section 5.10.3). English is the language most commonly evaluated (Section 2.3), but based on its typological and distributional properties English clearly is on the lower side of the spectrum (Figure 5.2), with most languages being more morphologically-complex, and thereby featuring more low-frequency words which are less likely to be learned well with current modeling approaches (Table 5.7).

The results of Part II and Part III thereby point without doubt to the immense limitations of purely context window based training. Word representations trained exclusively based on co-occurrence in large corpora will learn a representation expressing very general word meaning, which has shown helpful in many applications over previous methodology. However, at the

same time, these approaches will not only conflate multiple useful relations, but are also unable to accurately model low frequency phenomena, both considering semantic relations (Figure 3.2a) and the distributional properties of many of the world's languages (Sections 5.3, 5.10.3). As these limitations largely seem to arise from a combination of the model architecture, training and data sparsity, we think it is crucial for next-generation solutions to explore approaches that can efficiently leverage small data, as well as allow for a more fine-grained model of language, beyond semantic similarity.

This dissertation contributes a few modeling approaches going precisely into this direction. We have seen that it is possible to learn a mapping function from semantic similarity to entailment given a distributional vector space and a set of graded scores for lexical entailment (Section 4.6.1, Table 4.7). Further, we showed it is possible to fine-tune low-frequency word representations, leading to performance improvements especially in morphologically-rich languages (Table 5.7). Our approach enriches the word vectors with subword-level information to capture similar character sequences and consequently facilitates word-level language modeling prediction. The method has been implemented as additional fine-tuning step after each epoch of language model training, and the subword-level knowledge is extracted in an unsupervised manner from earlier character-aware layers of the same model. The approach leads to especially large improvements for morphologically-rich languages, and tackles precisely the data sparsity issues occurring due to their high morphological complexity.

Finally, we have introduced the idea of learning function-specific word representations in Chapter 6, and have demonstrated the usefulness of such specialised spaces in modeling a prominent linguistic structure (subject-verb-object). The model achieves state-of-the-art performance on a number of tasks reasoning over the SVO structure, which previously has only been achieved with much larger task-specific models (Section 6.5.1). The model creates a joint vector space for words grouped by type (e.g. subject, verb and object). It looks at all combinations between those groups (i.e. SV, VO, SO), and optimises the space such that vectors for plausible word combinations will lie close in a joint space (e.g. the verb vector *drink* will be close to a plausible object vector such as *beer*). In other words, the model trains a joint word vector space for groups of words based on their mutual associations. The resulting vectors are highly flexible, and can be used both in composition (e.g. for event similarity) as well as separately to predict common associations (e.g. for selectional preference). As the model is not tied to the distributional hypothesis, but instead merely relies on data providing associations or co-occurrences of words divided into sensible groups, it can be applied to a diverse set of phenomena (Future Work).

In summary, this dissertation has looked at a complementary spectrum of semantic relations (similarity, entailment, selectional preference), and also considered the influence of typological variations across the world's languages. We found that distributional approaches to representation learning are limited in terms of model granularity and capacity to model a diverse set of semantic relations, as well as suffer heavily from data sparsity issues, which we are present for semantic relations and even more pronounced in morphologically-rich languages. We contributed novel evaluation sets as well as modeling approaches targeting these under-resourced phenomena.

In conclusion, data sparsity issues persist and are especially strong in morphologically-rich languages. This calls for an increased focus on approaches that can integrate information from different levels of processing (character, subword, word and phrase-level), as well as efficiently make use of data available. Furthermore, the findings of this dissertation point out that it might be necessary to rethink approaches to modeling word semantics. We have seen it is possible to model word representations for specific semantic relations, either by learning a mapping function from a general distributional space (Section 4.7), or by training representations from scratch with selected data explicitly targeting one phenomenon in particular (Chapter 6). However, the resulting relation-specific or function-specific spaces require a very different order from general semantic spaces (Sections 4.4.3, 6.3). Nevertheless, all of the above relations are important to human language understanding, and therefore without doubt a computational model of language should equally be able to reason over all of them. We call for next-level architectures that can efficiently leverage and integrate information about different semantic relations, across languages.

We hope this dissertation can serve as a valuable contribution to representation learning, and inspire future work in modeling semantic similarity and beyond.

# Future Work

In this thesis we have mainly focused on word-level representation learning, with the aim to contribute in particular to previously under-resourced phenomena in terms of semantic relations and linguistic diversity.

In Part II we contributed novel evaluations verb similarity and entailment. Chapter 3 introduced an intrinsic evaluation set for verb similarity, SimVerb-3500 (Gerz et al., 2016). The data set has since been used for evaluating new models (Collell et al., 2017; Emerson and Copestake, 2017; Mrkšić et al., 2017, inter alia), and helped to ensure that verbs are adequately represented. Especially with the increased interest in vector space specialisation to mitigate some of the limitations of purely distributional training (Ponti et al., 2018b; Rei et al., 2018; Mrkšić et al., 2017), SimVerb helps by ensuring good performance on verbs. Further, SimVerb has inspired a data set for the biomedical domain: Bio-SimVerb (Chiu et al., 2018). Similar to SimVerb, HyperLex discussed in Chapter 4, has been used to develop new representation models specific to the relation of entailment (Nickel and Kiela, 2017; Vulić and Mrkšić, 2018; Roller et al., 2018, inter alia)

In Part III we looked at language modeling and highlighted the need for next-generation architectures to focus on the challenging multilingual setting, where data sparsity issues might prevent learning of high-quality word representations. Interest in typological comparisons is increasing recently (Bjerva and Augenstein, 2018; Ponti et al., 2018a; Bjerva et al., 2019, inter alia), and similar ideas combining word with character-level information are being explored (Schick and Schütze, 2018). Further, we have since contributed to a benchmark of state-of-the-art (Baysian) n-gram modeling approaches in Shareghi et al. (2019), which validates our evaluation setup.

Future work may combine these directions, and investigate semantic relations with morphologically-rich languages in mind. Both SimVerb and SimLex, as well as their multilingual variants (Leviant and Reichart, 2015), concentrate on the dictionary forms of words. Novel datasets designed especially for evaluating models of morphologically-rich languages may investigate semantic relations such as semantic similarity, entailment, and selectional preference between different morphological variants.

Recently, the field of NLP has seen an increased interest especially into context-aware (Peters et al., 2018; Devlin et al., 2019; Howard and Ruder, 2018b) as well as sentence representations (Cer et al., 2018; Yang et al., 2018). One straightforward step is to consider the findings of this dissertation in the light of these novel representation learning approaches. While current context-aware models can produce one vector per phrase or sentence, the architectures might still rely on word representations (e.g. ELMO by Peters et al. (2018) using the architecture by Kim et al. (2016) which we have looked at in Part III). Subword-level representations can partially mitigate this issue, but the exact influence of pure word-level versus pure subword-level representations on performance remain unclear (Jozefowicz et al., 2016; Schick and Schütze, 2018). While subword-level representations do partially mitigate data sparsity issues, they do so at the expense of removing model parameters dedicated to a higher-level semantic unit (i.e. a word or bigram) (Zhu et al., 2019). It is challenging to draw exact comparisons between architectures, because the standard evaluation metric (perplexity) inherently depends on the vocabulary size of the model (Section 2.3). Fair comparisons especially between models of drastically different vocabulary sizes (character, subword and word-level approaches) are not possible in this framework, calling for the design of novel metrics.

Another important aspect is that even context-aware models on large amounts of data still cannot model low-frequency words (Schick and Schütze, 2019) and likely still struggle to provide adequate flexibility to learn multiple semantic relations without conflating them. We therefore believe it is crucial to keep expanding evaluations to both more semantic as well as syntactic relations, spanning an increasingly diverse set of languages. Thanks to the ability of novel approaches to learn and construct representations for words, phrases or sentences, new evaluations can also cover a much broader range and include sentence-level phenomena as well (Conneau and Kiela, 2018; Wang et al., 2018, 2019). We believe having more broad-coverage phrase or sentence-level evaluations of a diverse range of relations and languages will be immensely useful.

Further, we believe there will be a continued need for investigating possible ways forward to mitigate data sparsity and few shot learning. One way could be to apply and test recent advances in embedding compression. Architectures such as Slim (Li et al., 2018) and WEST (Variani et al., 2019) do not train a single word embedding, but construct it efficiently from several subvectors, showing promise both in terms of run time as well as for being more robust to data sparsity issues. This strand of models predicts words, but constructs the next-word prediction from subword information, and is thereby able to achieve a reliable next-word prediction while being fully aware of subword units. Variani et al. (2019) use a fixed number

of splits per word for efficiency reasons. An approach for morphologically-rich languages could expand this idea to a variable number of splits corresponding to relevant morphemes.

In addition, with the rise of sentence encoders and increased interest in conversational agents (Henderson et al., 2019), fine-grained knowledge of semantic and syntactic relations, as well as incorporating real-world knowledge into neural models becomes increasingly important. We hope that especially our work from Chapter 6 on function-specific spaces can inspire future work in the area. The model can be applied to many different relations, both syntactic such as *adjective-noun* and semantic, such as for transforming domain-specific knowledge into vector space format (e.g. dishes, ingredients, cuisine for restaurants), thus making it accessible to neural models. While chapter 6 has primarily looked at relations standing in a fully-connected relationship, a logical next step for the methodology would be to consider a partially-connected setup. Some semantic or syntactic structures might not require taking into account all directions, and leaving out unnecessary directions will increase computational efficiency. One way to incorporate function-specific embeddings into sentence representations could for example be to make use of recent advances in memory-based neural architectures (Weston et al., 2015; Graves et al., 2014, 2016), which have shown it is possible for a neural network to read and write to vector representations. We envision a novel line of such memory-equipped models able to access pre-trained relation-specific and function-specific semantic knowledge, thus further mitigating limitations of purely distributional training and data sparsity issues.

Finally, and as one of our main conclusions and calls for future work, we believe it is absolutely crucial to consider and learn from recent findings in neuroscience and cognitive science. We have found these immensely helpful both to guide the development of our evaluation sets (Part II), as well as for the design of the function-specific model (Part IV). In image processing we have seen important advances driven by research on visual processing in the human brain (LeCun et al., 1990). While the neurobiology of language is still far less understood, there exists a number of known aspects not widely considered for designing NLP models. One such aspect is that on the word-level, humans generally seem to make a distinction between a few fundamental categories such as *concrete* vs. *abstract*, *animate* vs. *inanimate* as well as objects taken from the *natural* world vs. *artificial* objects (e.g. in modern cities) (Caramazza and Shelton, 1998; Huth et al., 2012, 2016, inter alia). Further, there are a number of findings regarding the organisation of language in the brain that hypothesize a mixture of both category and modality-specific distributed representations, combined through hubs (Hickok and Small, 2015, Chapters 50, 61). It seems worth trying to experiment with such hub-based structures in our artificial neural networks. Finally, most recent studies correlate vector and matrix representations extracted from brain scans to

standard NLP representations and models trained from corpora. For instance, comparing a recurrent neural network language model to brain activations has highlighted that different regions in the brain correlate to sequences of different length (Jain and Huth, 2018). This indicates it might be beneficial to explore architectures equipped with the flexibility to process short or mid-length sequences in distinct ways, rather than having one sequentially updated memory cell (Hochreiter and Schmidhuber, 1997). In summary, looking at neuroscience for guidance seems like a particularly exciting and promising future direction to us, particularly as models trained from data are increasingly tested for correlation to brain scans (Huth et al., 2012, 2016; Jain and Huth, 2018). We believe increased exploration of artificial models in conjunction with neuroscience has the potential to contribute to both to our understanding of the brain, as well as highlight aspects important to language understanding that are currently not present in our artificial models of language.

# Bibliography


Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283.

Adams, O., Makarucha, A., Neubig, G., Bird, S., and Cohn, T. (2017). Cross-lingual word embeddings for low-resource language modeling. In *Proceedings of EACL*, pages 937–947.

Agirre, E., Alfonseca, E., Hall, K. B., Kravalova, J., Pasca, M., and Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of NAACL-HLT*, pages 19–27.

Al-Rfou, R., Perozzi, B., and Skiena, S. (2013). Polyglot: Distributed word representations for multilingual NLP. In *Proceedings of CoNLL*, pages 183–192.

Anderson, A., Kiela, D., Clark, S., and Poesio, M. (2017). Visually grounded and textual semantic models differentially decode brain activity associated with concrete and abstract nouns. *Transactions of the ACL*, 5:17–30.

Astudillo, R., Amir, S., Ling, W., Silva, M., and Trancoso, I. (2015). Learning word representations from scarce and noisy data with embedding subspaces. In *Proceedings of ACL*, pages 1074–1084.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet project. In *ACL-COLING*, pages 86–90.

Baker, S., Reichart, R., and Korhonen, A. (2014). An Unsupervised Model for Instance Level Subcategorization Acquisition. In *EMNLP*, pages 278–289.

Bakker, D. (2010). Language sampling. In *The Oxford handbook of linguistic typology*, pages 100–127. Oxford University Press.

Baroni, M., Bernardi, R., Do, N.-Q., and Shan, C.-c. (2012). Entailment above the word level in distributional semantics. In *Proceedings of EACL*, pages 23–32.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (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 *ACL*, pages 238–247.

Baroni, M. and Lenci, A. (2010). Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Baroni, M. and Lenci, A. (2011). How we BLESSed distributional semantic evaluation. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics (GEMS)*, pages 1–10.

Beckwith, R., Fellbaum, C., Gross, D., and Miller, G. A. (1991). WordNet: A lexical database organized on psycholinguistic principles. *Lexical acquisition: Exploiting on-line resources to build a lexicon*, pages 211–231.

Beltagy, I., Chau, C., Boleda, G., Garrette, D., Erk, K., and Mooney, R. (2013). Montague meets Markov: Deep semantics with probabilistic logical form. In *Proceedings of *SEM*, pages 11–21.

Bender, E. M. (2013). *Linguistic fundamentals for natural language processing: 100 essentials from morphology and syntax*. Morgan & Claypool Publishers.

Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Bergmanis, T., Kann, K., Schütze, H., and Goldwater, S. (2017). Training data augmentation for low-resource morphological inflection. In *Proceedings of CoNLL*, pages 31–39.

Bergsma, S., Lin, D., and Goebel, R. (2008). Discriminative learning of selectional preference from unlabeled text. In *Proceedings of EMNLP*, pages 59–68.

Bickel, B. and Nichols, J. (2013). *Inflectional Synthesis of the Verb*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Binder, J. R., Desai, R. H., Graves, W. W., and Conant, L. L. (2009). Where Is the Semantic System? A Critical Review and Meta-Analysis of 120 Functional Neuroimaging Studies. *Cerebral Cortex*, 19(12):2767–2796.

Birch, A., Osborne, M., and Koehn, P. (2008). Predicting success in machine translation. In *Proceedings of EMNLP*, pages 745–754.

Bird, S. (2011). Bootstrapping the language archive: New prospects for natural language processing in preserving linguistic heritage. *Linguistic Issues in Language Technology*, 6(4).

Bjerva, J. and Augenstein, I. (2018). From phonology to syntax: Unsupervised linguistic typology at different levels with language embeddings. 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 907–916, New Orleans, Louisiana. Association for Computational Linguistics.

Bjerva, J., Östling, R., Han Veiga, M., Tiedemann, J., and Augenstein, I. (2019). What do language representations really represent? *Computational Linguistics*, (Just Accepted):1–8.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the ACL*, 5:135–146.

Bojar, O., Buck, C., Callison-Burch, C., Federmann, C., Haddow, B., Koehn, P., Monz, C., Post, M., Soricut, R., and Specia, L. (2013). Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the 8th Workshop on Statistical Machine Translation*, pages 1–44.

Bornkessel-Schlesewsky, I. and Schlesewsky, M. (2016). The argument dependency model. In *Neurobiology of language*, pages 357–369. Elsevier.

Bos, J. and Markert, K. (2005). Recognising textual entailment with logical inference. In *Proceedings of EMNLP*, pages 628–635.

Botha, J. A. and Blunsom, P. (2014). Compositional morphology for word representations and language modelling. In *Proceedings of ICML*, pages 1899–1907.

Bruni, E., Tran, N.-K., and Baroni, M. (2014). Multimodal Distributional Semantics. *Journal of Artificial Intelligence Research*, 49:1–47.

Camacho-Collados, J., Pilehvar, M. T., and Navigli, R. (2015). A framework for the construction of monolingual and cross-lingual word similarity datasets. In *Proceedings of ACL*, pages 1–7.

Caramazza, A. and Hillis, A. E. (1991). Lexical organization of nouns and verbs in the brain. *Nature*, 349(6312):788–790.

Caramazza, A. and Shelton, J. R. (1998). Domain-Specific Knowledge Systems in the Brain: The Animate-Inanimate Distinction. *Journal of Cognitive Neuroscience*, 10(1):1–34.

Cer, D., Yang, Y., yi Kong, S., Hua, N., Limtiaco, N., John, R. S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y.-H., Strope, B., and Kurzweil, R. (2018). Universal sentence encoder. *CoRR*, abs/1803.11175.

Chambers, N. and Jurafsky, D. (2009). Unsupervised learning of narrative schemas and their participants. In *Proceedings of ACL*, pages 602–610.

Chambers, N. and Jurafsky, D. (2010). Improving the use of pseudo-words for evaluating selectional preferences. In *Proceedings of ACL*, pages 445–453.

Chatfield, K., Simonyan, K., Vedaldi, A., and Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. In *Proceedings of BMVC*.

Chelba, C., Mikolov, T., Schuster, M., Ge, Q., Brants, T., and Koehn, P. (2013). One billion word benchmark for measuring progress in statistical language modeling. *CoRR*, abs/1312.3005.

Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of EMNLP*, pages 740–750.

Chen, S. F. and Goodman, J. (1999). An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, 13(4):359–394.

Chen, X., Liu, X., Qian, Y., Gales, M., and Woodland, P. C. (2016). CUED-RNNLM: An open-source toolkit for efficient training and evaluation of recurrent neural network language models. In *Proceedings of ICASSP*, pages 6000 –6004.

Chiu, B., Korhonen, A., and Pyysalo, S. (2016). Intrinsic Evaluation of Word Vectors Fails to Predict Extrinsic Performance. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 1–6, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chiu, B., Pyysalo, S., Vulić, I., and Korhonen, A. (2018). Bio-simverb and bio-simlex: wide-coverage evaluation sets of word similarity in biomedicine. *BMC bioinformatics*, 19(1):33.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *EMNLP*.

Coecke, B., Sadrzadeh, M., and Clark, S. (2010). Mathematical foundations for a compositional distributional model of meaning. *Linguistic Analysis*, 36(1-4):345–384.

Coleman, L. and Kay, P. (1981). Prototype semantics: The English word lie. *Language*, 57(1):26–44.

Collell, G., Zhang, T., and Moens, M.-F. (2017). Imagined visual representations as multimodal embeddings. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Collins, A. M. and Quillian, R. M. (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, 8(2):240–247.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: deep neural networks with multitask learning. In *ICML*, pages 160–167.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Conneau, A. and Kiela, D. (2018). Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*.

Cotterell, R. and Eisner, J. (2017). Probabilistic typology: Deep generative models of vowel inventories. In *Proceedings of ACL*, pages 1182–1192.

Cotterell, R., Mielke, S. J., Eisner, J., and Roark, B. (2018). Are all languages equally hard to language-model? In *Proceedings of NAACL-HLT*.

Cotterell, R., Müller, T., Fraser, A., and Schütze, H. (2015). Labeled morphological segmentation with semi-Markov models. In *Proceedings of CoNLL*, pages 164–174.

Cotterell, R., Schütze, H., and Eisner, J. (2016). Morphological smoothing and extrapolation of word embeddings. In *Proceedings of ACL*, pages 1651–1660.

Dagan, I., Glickman, O., and Magnini, B. (2006). The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges*, pages 177–190.

Daiber, J. (2018). *Typologically Robust Statistical Machine Translation*. PhD thesis, University of Amsterdam.

Damasio, A. R. and Tranel, D. (1993). Nouns and verbs are retrieved with differently distributed neural systems. *Proceedings of the National Academy of Sciences of the United States of America*, 90(11):4957–60.

Davies, R. R., Halliday, G. M., Xuereb, J. H., Kril, J. J., and Hodges, J. R. (2009). The neural basis of semantic memory: Evidence from semantic dementia. *Neurobiology of Aging*, 30(12):2043–2052.

Davis, M. H. (2016). The neurobiology of lexical access. In *Neurobiology of language*, pages 541–555. Elsevier.

de Heer, W. A., Huth, A. G., Griffiths, T. L., Gallant, J. L., and Theunissen, F. E. (2017). The hierarchical cortical organization of human speech processing. *Journal of Neuroscience*, 37(27):6539–6557.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Deschacht, K. and Moens, M. (2009). Semi-supervised semantic role labeling using the latent words language model. In *Proceedings of EMNLP*, pages 21–29.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

Dryer, M. S. (1989). Large linguistic areas and language sampling. *Studies in Language. International Journal sponsored by the Foundation "Foundations of Language"*, 13(2):257–292.

Dryer, M. S. and Haspelmath, M., editors (2013). *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Edelstein, L. and Reichart, R. (2016). A factorized model for transitive verbs in compositional distributional semantics. *CoRR*, abs/1609.07756.

Emerson, G. and Copestake, A. (2017). Semantic composition via probabilistic model theory. *In Proceedings of the 12th International Conference on Computational Semantics (IWCS)*.

Emerson, G. and Copestake, A. A. (2016). Functional distributional semantics. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 40–52.

Erk, K., Padó, S., and Padó, U. (2010). A flexible, corpus-driven model of regular and inverse selectional preferences. *Computational Linguistics*, 36(4):723–763.

Evans, N. and Levinson, S. C. (2009). The myth of language universals: Language diversity and its importance for cognitive science. *Behavioral and Brain Sciences*, 32(5):429–448.

Faruqui, M. (2016). *Diverse Context for Learning Word Representations*. PhD thesis, Carnegie Mellon University.

Faruqui, M., Dodge, J., Jauhar, S. K., Dyer, C., Hovy, E. H., and Smith, N. A. (2015). Retrofitting word vectors to semantic lexicons. In *NAACL-HLT*, pages 1606–1615.

Faruqui, M. and Dyer, C. (2015). Non-distributional Word Vector Representations. In *ACL*, pages 464–469.

Faruqui, M., Tsvetkov, Y., Rastogi, P., and Dyer, C. (2016). Problems With Evaluation of Word Embeddings Using Word Similarity Tasks. *CoRR*, abs/1605.0.

Fellbaum, C. (1998). *WordNet*.

Filippova, K., Alfonseca, E., Colmenares, C. A., Kaiser, L., and Vinyals, O. (2015). Sentence compression by deletion with LSTMs. In *Proceedings of EMNLP*, pages 360–368.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Firth, J. R. (1968). *Selected papers of JR Firth, 1952-59*. Indiana University Press.

Gandhe, A., Metze, F., and Lane, I. (2014). Neural network language models for low resource languages. In *Proceedings of INTERSPEECH*, pages 2615–2619.

Ganitkevitch, J., Van Durme, B., and Callison-Burch, C. (2013). PPDB: The Paraphrase Database. In *NAACL-HLT*, pages 758–764.

Gell-Mann, M. and Ruhlen, M. (2011). The origin and evolution of word order. *Proceedings of the National Academy of Sciences*, 108(42):17290–17295.

Gentner, D. (2006). Why verbs are hard to learn. *Action meets word: How children learn verbs*, pages 544–564.

Gerz, D., Vulić, I., Hill, F., Reichart, R., and Korhonen, A. (2016). Simverb-3500: A large-scale evaluation set of verb similarity. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2173–2182, Austin, Texas. Association for Computational Linguistics.

Gladkova, A. and Drozd, A. (2016). Intrinsic evaluations of word embeddings: What can we do better? In *Proceedings of REPEVAL*, pages 36–42.

Goldberg, Y. and Levy, O. (2014). word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722.

Goodman, J. T. (2001). A bit of progress in language modeling. *Computer Speech & Language*, 15(4):403–434.

Grave, E., Cissé, M., and Joulin, A. (2017). Unbounded cache model for online language modeling with open vocabulary. In *Proceedings of NIPS*, pages 6044–6054.

Graves, A. (2013). Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850.

Graves, A., Wayne, G., and Danihelka, I. (2014). Neural turing machines. *arXiv preprint arXiv:1410.5401*.

Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., Colmenarejo, S. G., Grefenstette, E., Ramalho, T., Agapiou, J., et al. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471.

Greenberg, C., Demberg, V., and Sayeed, A. (2015a). Verb polysemy and frequency effects in thematic fit modeling. In *Proceedings of the 6th Workshop on Cognitive Modeling and Computational Linguistics*, pages 48–57.

Greenberg, C., Sayeed, A., and Demberg, V. (2015b). Improving unsupervised vector-space thematic fit evaluation via role-filler prototype clustering. In *Proceedings of NAACL-HLT*, pages 21–31.

Greene, D. and Cunningham, P. (2006). Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of ICML*, pages 377–384.

Grefenstette, E. and Sadrzadeh, M. (2011a). Experimental support for a categorical compositional distributional model of meaning. In *Proceedings of EMNLP*, pages 1394–1404.

Grefenstette, E. and Sadrzadeh, M. (2011b). Experimenting with transitive verbs in a DisCoCat. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 62–66.

Hampton, J. A. (2007). Typicality, graded membership, and vagueness. *Cognitive Science*, 31(3):355–384.

Harris, Z. S. (1954). Distributional Structure. *Word*, 10(2-3):146–162.

Hashimoto, K. and Tsuruoka, Y. (2016). Adaptive joint learning of compositional and non-compositional phrase embeddings. In *Proceedings of ACL*, pages 205–215.

Hashimoto, T. B., Alvarez-Melis, D., and Jaakkola, T. S. (2016). Word embeddings as metric recovery in semantic spaces. *Transactions of the ACL*, 4:273–286.

Haspelmath, M. and Sims, A. (2013). *Understanding morphology*.

Heafield, K., Pouzyrevsky, I., Clark, J. H., and Koehn, P. (2013). Scalable modified Kneser-Ney language model estimation. In *Proceedings of ACL*, pages 690–696.

Heaps, H. S. (1978). *Information retrieval, computational and theoretical aspects*. Academic Press.

Henderson, M., Al-Rfou, R., Strope, B., Sung, Y.-h., Luk'acs, L., Guo, R., Kumar, S., Miklos, B., and Kurzweil, R. (2017). Efficient Natural Language Response Suggestion for Smart Reply. *ArXiv e-prints*.

Henderson, M., Budzianowski, P., Casanueva, I., Coope, S., Gerz, D., Kumar, G., Mrksic, N., Spithourakis, G., hao Su, P., Vulic, I., and Wen, T.-H. (2019). A repository of conversational datasets. *CoRR*, abs/1904.06472.

Herbelot, A. and Baroni, M. (2017). High-risk learning: acquiring new word vectors from tiny data. In *Proceedings of EMNLP*, pages 304–309.

Herbelot, A. and Ganesalingam, M. (2013). Measuring semantic content in distributional vectors. In *Proceedings of ACL*, pages 440–445.

Herdan, G. (1960). *Type-token mathematics*, volume 4. Mouton.

Hickok, G. and Small, S. (2015). *Neurobiology of Language*. Elsevier, The address, 1 edition.

Hill, F., Reichart, R., and Korhonen, A. (2015). SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc.

Howard, J. and Ruder, S. (2018a). Universal language model fine-tuning for text classification. In *Proceedings of ACL*, pages 328–339.

Howard, J. and Ruder, S. (2018b). Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339.

Huth, A. G., de Heer, W. A., Griffiths, T. L., Theunissen, F. E., and Gallant, J. L. (2016). Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature*, 532(7600):453–458.

Huth, A. G., Nishimoto, S., Vu, A. T., and Gallant, J. L. (2012). A continuous semantic space describes the representation of thousands of object and action categories across the human brain. *Neuron*, 76(6):1210–1224.

Jain, S. and Huth, A. (2018). Incorporating context into language encoding models for fmri. In *Advances in Neural Information Processing Systems*, pages 6628–6637.

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., and Wu, Y. (2016). Exploring the limits of language modeling. In *Proceedings of ICML*.

Jurafsky, D. and Martin, J. H. (2017). *Speech and Language Processing*, volume 3. Pearson.

Kamp, H. and Partee, B. (1995). Prototype theory and compositionality. *Cognition*, 57(2):129–191.

Kartsaklis, D. and Sadrzadeh, M. (2014). A study of entanglement in a categorical framework of natural language. In *Proceedings of QPL*, pages 249–261.

Kartsaklis, D., Sadrzadeh, M., and Pulman, S. (2012). A unified sentence space for categorical distributional-compositional semantics: Theory and experiments. In *Proceedings of COLING*, pages 549–558.

Kawakami, K., Dyer, C., and Blunsom, P. (2017). Learning to create and reuse words in open-vocabulary neural language modeling. In *Proceedings of ACL*, pages 1492–1502.

Kiela, D., Hill, F., and Clark, S. (2015a). Specializing word embeddings for similarity or relatedness. In *Proceedings of EMNLP*, pages 2044–2048.

Kiela, D., Rimell, L., Vulić, I., and Clark, S. (2015b). Exploiting image generality for lexical entailment detection. In *ACL*, pages 119–124.

Kilgarriff, A. (1997). Putting Frequencies in the Dictionary. *International Journal of Lexicography*, 10(2):135–155.

Kim, Y., Jernite, Y., Sontag, D., and Rush, A. M. (2016). Character-aware neural language models. *Proceedings of AAAI*, pages 2741–2749.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of ICLR (Conference Track)*.

Kipper, K., Korhonen, A., Ryant, N., and Palmer, M. (2008). A large-scale classification of English verbs. *Language Resource and Evaluation*, 42(1):21–40.

Kipper, K., Snyder, B., and Palmer, M. (2004). Extending a Verb-lexicon Using a Semantically Annotated Corpus. In *LREC*, pages 1557–1560.

Kneser, R. and Ney, H. (1995). Improved backing-off for M-gram language modeling. In *Proceedings of ICASSP*, pages 181–184.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of the 10th Machine Translation Summit*, pages 79–86.

Kotlerman, L., Dagan, I., Szpektor, I., and Zhitomirsky-Geffet, M. (2010). Directional distributional similarity for lexical inference. *Natural Language Engineering*, 16(4):359–389.

Lake, B. M., Salakhutdinov, R., Gross, J., and Tenenbaum, J. B. (2011). One shot learning of simple visual concepts. In *CogSci*.

Lakoff, G. (1990). *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*.

Landauer, T. K., Foltz, P. W., and Laham, D. (1998). An introduction to latent semantic analysis. *Discourse processes*, 25(2-3):259–284.

Leacock, C. and Chodorow, M. (1998). Combining local context and WordNet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283.

LeCun, Y., Boser, B., Denker, J. S., Howard, R. E., Habbard, W., Jackel, L. D., and Henderson, D. (1990). Advances in neural information processing systems 2. chapter Handwritten Digit Recognition with a Back-propagation Network, pages 396–404. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Lee, K., Levy, O., and Zettlemoyer, L. (2017). Recurrent additive networks. *CoRR*, abs/1705.07393.

Leech, G. N. (1992). 100 million words of English: The British National Corpus (BNC).

Leviant, I. and Reichart, R. (2015). Separated by an un-common language: Towards judgment language informed vector space modeling.

Levin, B. (1993). *English verb classes and alternation, A preliminary investigation*. The University of Chicago Press.

Levy, O. and Goldberg, Y. (2014a). Dependency-based word embeddings. In *Proceedings of ACL*, pages 302–308.

Levy, O. and Goldberg, Y. (2014b). Neural word embedding as implicit matrix factorization. In *Proceedings of NIPS*, pages 2177–2185.

Levy, O., Goldberg, Y., and Dagan, I. (2015a). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the ACL*, 3:211–225.

Levy, O., Remus, S., Biemann, C., and Dagan, I. (2015b). Do supervised distributional methods really learn lexical inference relations? In *NAACL-HLT*, pages 970–976.

Li, Z., Kulhanek, R., Wang, S., Zhao, Y., and Wu, S. (2018). Slim embedding layers for recurrent neural language models. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Ling, W., Luís, T., Marujo, L., Astudillo, R. F., Amir, S., Dyer, C., Black, A. W., and Trancoso, I. (2015). Finding function in form: Compositional character models for open vocabulary word representation. *Proceedings of EMNLP*, pages 1520–1530.

Liu, Q., Jiang, H., Wei, S., Ling, Z.-H., and Hu, Y. (2015). Learning semantic word embeddings based on ordinal knowledge constraints. In *Proceedings of ACL*, pages 1501–1511.

Loper, E., Yi, S.-T., and Palmer, M. (2007). Combining lexical resources: Mapping between propbank and verbnet. In *IWCS*.

Luong, M.-T. and Manning, C. D. (2016). Achieving open vocabulary neural machine translation with hybrid word-character models. In *Proceedings of ACL*, pages 1054–1063.

Luong, T., Socher, R., and Manning, C. (2013). Better Word Representations with Recursive Neural Networks for Morphology. In *CoNLL*, pages 104–113.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of ACL*, pages 142–150.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Markman, A. B. and Wisniewski, E. J. (1997). Similar and different: The differentiation of basic-level categories. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 23(1).

McCarthy, R. A. and Warrington, E. K. (1988). Evidence for modality-specific meaning systems in the brain. *Nature*, 334(6181):428–430.

McRae, K., Ferretti, T., and Amyote, L. (1997). Thematic roles as verb-specific concepts. *Language and Cognitive Processes*, 12(2):137–176.

McRae, K., Khalkhali, S., and Hare, M. (2012). Semantic and associative relations in adolescents and young adults: Examining a tenuous dichotomy.

McRae, K., Spivey-Knowlton, M. J., and Tanenhaus, M. K. (1998). Modeling the influence of thematic fit (and other constraints) in on-line sentence comprehension. *Journal of Memory and Language*, 38(3):283–312.

McWhorter, J. (2001). The world's simplest grammars are Creole grammars. *Linguistic Typology*, 5(2):125–66.

Medin, D. L., Altom, M. W., and Murphy, T. D. (1984). Given versus induced category representations: Use of prototype and exemplar information in classification. *Journal of Experimental Psychology*, 10(3):333–352.

Melamud, O., McClosky, D., Patwardhan, S., and Bansal, M. (2016). The role of context types and dimensionality in learning word embeddings. In *Proceedings of NAACL-HLT*, pages 1030–1040.

Meyer, L. and Friederici, A. D. (2016). Neural systems underlying the processing of complex sentences. In *Neurobiology of language*, pages 597–606. Elsevier.

Mikolov, T., Chen, K., Corrado, G. S., and Dean, J. (2013a). Efficient estimation of word representations in vector space. In *Proceedings of ICLR (Workshop Papers)*.

Mikolov, T., Karafiát, M., Burget, L., Cernockỳ, J., and Khudanpur, S. (2010). Recurrent neural network based language model. In *Proceedings of INTERSPEECH*, pages 1045–1048.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 27th Annual Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119.

Milajevs, D., Kartsaklis, D., Sadrzadeh, M., and Purver, M. (2014). Evaluating neural word representations in tensor-based compositional settings. In *Proceedings of EMNLP*, pages 708–719.

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41.

Mitchell, J. and Lapata, M. (2008). Vector-based models of semantic composition. *Proceedings of ACL*, pages 236–244.

Mitchell, T. M., Shinkareva, S. V., Carlson, A., Chang, K.-M., Malave, V. L., Mason, R. A., and Just, M. A. (2008). Predicting human brain activity associated with the meanings of nouns. *Science*, 320(5880):1191–1195.

Miyamoto, Y. and Cho, K. (2016). Gated word-character recurrent language model. In *Proceedings of EMNLP*, pages 1992–1997.

Modi, A. (2016). Event embeddings for semantic script modeling. In *Proceedings of CoNLL*, pages 75–83.

Mrkšić, N., Séaghdha, D. Ó., Thomson, B., Gašić, M., Rojas-Barahona, L. M., Su, P., Vandyke, D., Wen, T., and Young, S. J. (2016). Counter-fitting word vectors to linguistic constraints. In *Proceedings of NAACL-HLT*.

Mrkšić, N., Séaghdha, D. Ó., Wen, T., Thomson, B., and Young, S. J. (2017). Neural belief tracker: Data-driven dialogue state tracking. In *Proceedings of ACL*.

Mrkšić, N., Vulić, I., Ó Séaghdha, D., Leviant, I., Reichart, R., Gašić, M., Korhonen, A., and Young, S. (2017). Semantic specialisation of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the ACL*, 5:309–324.

Müller, T., Schütze, H., and Schmid, H. (2012). A comparative investigation of morphological language modeling for the languages of the European Union. In *Proceedings of NAACL-HLT*, pages 386–395.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML 2010)*, pages 807–814.

Nelson, D. L., McEvoy, C. L., and Schreiber, T. A. (2004). The university of south florida free association, rhyme, and word fragment norms. *Behavior Research Methods*, 36(3):402–407.

Nickel, M. and Kiela, D. (2017). Poincaré embeddings for learning hierarchical representations. In *NIPS*, pages 6341–6350.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R. T., Petrov, S., Pyysalo, S., Silveira, N., et al. (2016). Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of LREC*, pages 1659–1666.

O'Horan, H., Berzak, Y., Vulić, I., Reichart, R., and Korhonen, A. (2016). Survey on the use of typological information in natural language processing. In *Proceedings of COLING*, pages 1297–1308.

Ono, M., Miwa, M., and Sasaki, Y. (2015). Word embedding-based antonym detection using thesauri and distributional information. In *Proceedings of NAACL-HLT*, pages 984–989.

Padó, S., Padó, U., and Erk, K. (2007). Flexible, Corpus-Based Modelling of Human Plausibility Judgements. In *EMNLP-CoNLL*, pages 400–409.

Padó, U. (2007). The integration of syntax and semantic plausibility in a wide-coverage model of human sentence processing.

Palmer, M., Kingsbury, P., and Gildea, D. (2005). The {Proposition Bank: A}n Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.

Pascual, B., Masdeu, J. C., Hollenbeck, M., Makris, N., Insausti, R., Ding, S.-L., and Dickerson, B. C. (2015). Large-scale brain networks of the human left temporal pole: A functional connectivity MRI study. *Cerebral Cortex*, 25(3):680–702.

Paul, M., Yamamoto, H., Sumita, E., and Nakamura, S. (2009). On the importance of pivot language selection for statistical machine translation. In *Proceedings of NAACL-HLT*, pages 221–224.

Pavlick, E., Rastogi, P., Ganitkevitch, J., Van Durme, B., and Callison-Burch, C. (2015). Ppdb 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *ACL*, pages 425–430.

Pedersen, T., Patwardhan, S., and Michelizzi, J. (2004). WordNet::Similarity - Measuring the relatedness of concepts. In *Proceedings of AAAI*, pages 1024–1025.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Pichotta, K. and Mooney, R. J. (2016). Learning statistical scripts with LSTM recurrent neural networks. In *Proceedings of AAAI*, pages 2800–2806.

Pilehvar, M. T., Kartsaklis, D., Prokhorov, V., and Collier, N. (2018). Card-660: A reliable evaluation framework for rare word representation models. In *EMNLP*, pages 1391–1401.

Pinter, Y., Guthrie, R., and Eisenstein, J. (2017). Mimicking word embeddings using subword RNNs. In *Proceedings of EMNLP*, pages 102–112.

Plank, F. (2017). Split morphology: How agglutination and flexion mix. *Linguistic Typology*, 21(2017):1–62.

Plaut, D. C. (1995). Semantic and associative priming in a distributed attractor network. In *Proceedings of the 17th annual conference of the cognitive science society*, volume 17, pages 37–42. Pittsburgh, PA.

Ponte, J. M. and Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proceedings of SIGIR*, pages 275–281.

Ponti, E. M., O'Horan, H., Berzak, Y., Vulić, I., Reichart, R., Poibeau, T., Shutova, E., and Korhonen, A. (2018a). Modeling language variation and universals: A survey on typological linguistics for natural language processing. *CoRR*, abs/1807.00914.

Ponti, E. M., Vulić, I., Glavaš, G., Mrkšić, N., and Korhonen, A. (2018b). Adversarial propagation and zero-shot cross-lingual transfer of word vector specialization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 282–293, Brussels, Belgium. Association for Computational Linguistics.

Ponti, E. M., Vulić, I., and Korhonen, A. (2017). Decoding sentiment from distributed representations of sentences. In *Proceedings of *SEM*, pages 22–32.

Press, O. and Wolf, L. (2017). Using the output embedding to improve language models. In *Proceedings of EACL*, pages 157–163.

Quillian, R. M. (1966). Semantic memory. Technical report, Bolt, Beranek and Newman.

Quillian, R. M. (1967). Word concepts: A theory and simulation of some basic semantic capabilities. *Behavioral Science*, 12(5):410–430.

Rei, M. (2017). Semi-supervised multitask learning for sequence labeling. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2121–2130, Vancouver, Canada. Association for Computational Linguistics.

Rei, M., Bulat, L., Kiela, D., and Shutova, E. (2017). Grasping the finer point: A supervised similarity network for metaphor detection. In *EMNLP*, pages 1537–1546.

Rei, M., Gerz, D., and Vulić, I. (2018). Scoring lexical entailment with a supervised directional similarity network. In *Proceedings of ACL*, pages 638–643.

Reisinger, J. and Mooney, R. J. (2010a). A Mixture Model with Sharing for Lexical Semantics. In *EMNLP*, pages 1173–1182.

Reisinger, J. and Mooney, R. J. (2010b). Multi-prototype vector-space models of word meaning. In *NAACL-HTL*, pages 109–117.

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of IJCAI*, pages 448–453.

Rice, G. E., Hoffman, P., and Lambon Ralph, M. A. (2015). Graded specialization within and between the anterior temporal lobes. *Annals of the New York Academy of Sciences*, 1359(1):84–97.

Riddoch, M. J., Humphreys, G. W., Coltheart, M., and Funnell, E. (1988). Semantic systems or system? Neuropsychological evidence re-examined. *Cognitive Neuropsychology*, 5(1):3–25.

Roller, S. and Erk, K. (2016). Relations such as hypernymy: Identifying and exploiting Hearst patterns in distributional vectors for lexical entailment. In *Proceedings of EMNLP*, pages 2163–2172.

Roller, S., Kiela, D., and Nickel, M. (2018). Hearst patterns revisited: Automatic hypernym detection from large text corpora. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 358–363, Melbourne, Australia. Association for Computational Linguistics.

Rooth, M., Riezler, S., Prescher, D., Carroll, G., and Beil, F. (1999). Inducing a semantically annotated lexicon via EM-based clustering. In *Proceedings of ACL*, pages 104–111.

Rosch, E. H. (1973). Natural categories. *Cognitive Psychology*, 4(3):328–350.

Rosch, E. H. (1975). Cognitive representations of semantic categories. *Journal of Experimental Psychology*, 104(3):192–233.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Rush, A. M., Chopra, S., and Weston, J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of EMNLP*, pages 379–389.

Sahlgren, M. (2008). The distributional hypothesis. *Italian Journal of Disability Studies*, 20:33–53.

Santus, E., Lenci, A., Lu, Q., and Schulte im Walde, S. (2014). Chasing hypernyms in vector spaces with entropy. In *EACL*, pages 38–42.

Sayeed, A., Greenberg, C., and Demberg, V. (2016). Thematic fit evaluation: An aspect of selectional preferences. In *Proceedings of the 1st Workshop on Evaluating Vector Space Representations for NLP*, pages 99–105.

Schick, T. and Schütze, H. (2018). Learning semantic representations for novel words: Leveraging both form and context. *arXiv preprint arXiv:1811.03866*.

Schick, T. and Schütze, H. (2019). Rare words: A major problem for contextualized embeddings and how to fix it by attentive mimicking.

Schnabel, T., Labutov, I., Mimno, D., and Joachims, T. (2015). Evaluation methods for unsupervised word embeddings. In *EMNLP*, pages 298–307.

Schütze, H. (1993). Word space. In *Proceedings of NIPS*, pages 895–902.

Schwartz, R., Reichart, R., and Rappoport, A. (2015). Symmetric pattern based word embeddings for improved word similarity prediction. In *Proceedings of CoNLL*, pages 258–267.

Serban, I. V., Sordoni, A., Bengio, Y., Courville, A. C., and Pineau, J. (2016). Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of AAAI*, pages 3776–3784.

Shareghi, E., Gerz, D., Vulić, I., and Korhonen, A. (2019). Show some love to your n-grams: A bit of progress and stronger n-gram language modeling baselines. In *Proceedings of NAACL-HLT*.

Shwartz, V., Goldberg, Y., and Dagan, I. (2016). Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of ACL*, pages 2389–2398.

Shwartz, V., Santus, E., and Schlechtweg, D. (2017). Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection. In *EACL*, pages 65–75.

Silberer, C. and Lapata, M. (2014). Learning Grounded Meaning Representations with Autoencoders. In *ACL*, pages 721–732.

Srivastava, R. K., Greff, K., and Schmidhuber, J. (2015). Highway networks. *ICML Deep Learning Workshop*.

Sundermeyer, M., Ney, H., and Schluter, R. (2015). From feedforward to recurrent LSTM neural networks for language modeling. *IEEE Transactions on Audio, Speech and Language Processing*, 23(3):517–529.

Sutskever, I., Vinyals, O., and Le, Q. (2014). Sequence to Sequence Learning with Neural Networks. *Advances in Neural Information Processing Systems*, pages 1–9.

Tiedemann, J. (2012). Parallel data, tools and interfaces in OPUS. In *Proceedings of LREC*, pages 2214–2218.

Tilk, O., Demberg, V., Sayeed, A., Klakow, D., and Thater, S. (2016). Event participant modelling with neural networks. In *Proceedings of EMNLP*, pages 171–182.

Tsvetkov, Y., Faruqui, M., Ling, W., Lample, G., and Dyer, C. (2015). Evaluation of Word Vector Representations by Subspace Alignment. In *EMNLP*, pages 2049–2054.

Turian, J. P., Ratinov, L., and Bengio, Y. (2010). Word representations: A simple and general method for semi-supervised learning. In *ACL*, pages 384–394.

Turney, P. D., Pantel, P., and others (2010). From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence research*.

Van de Cruys, T. (2009). A non-negative tensor factorization model for selectional preference induction. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, pages 83–90.

Van de Cruys, T. (2014). A neural network approach to selectional preference acquisition. In *Proceedings of EMNLP*, pages 26–35.

Vania, C. and Lopez, A. (2017). From characters to words to in between: Do we capture morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2016–2027, Vancouver, Canada. Association for Computational Linguistics.

Variani, E., Suresh, A. T., and Weintraub, M. (2019). West: Word encoded sequence transducers. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7340–7344. IEEE.

Vaswani, A., Zhao, Y., Fossum, V., and Chiang, D. (2013). Decoding with large-scale neural language models improves translation. In *Proceedings of EMNLP*, pages 1387–1392.

Vendrov, I., Kiros, R., Fidler, S., and Urtasun, R. (2016). Order-embeddings of images and language. In *Proceedings of ICLR*.

Vigliocco, G., Vinson, D. P., Druks, J., Barber, H., and Cappa, S. F. (2011). Nouns and verbs in the brain: a review of behavioural, electrophysiological, neuropsychological and imaging studies. *Neuroscience & Biobehavioral Reviews*, 35(3):407–426.

Vilnis, L. and McCallum, A. (2015). Word Representations via Gaussian Embedding. In *ICLR*, page 12.

Vulić, I., Gerz, D., Kiela, D., Hill, F., and Korhonen, A. (2017). Hyperlex: A large-scale evaluation of graded lexical entailment. *Computational Linguistics*, 43(4):781–835.

Vulić, I. and Mrkšić, N. (2018). Specialising word vectors for lexical entailment. 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 1134–1145, New Orleans, Louisiana. Association for Computational Linguistics.

Vulić, I., Mrkšić, N., Reichart, R., Ó Séaghdha, D., Young, S., and Korhonen, A. (2017). Morph-fitting: Fine-tuning word vector spaces with simple language-specific rules. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 56–68, Vancouver, Canada. Association for Computational Linguistics.

Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. (2019). Superglue: A stickier benchmark for general-purpose language understanding systems.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. (2018). Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355. Association for Computational Linguistics.

Wang, T. and Cho, K. (2016). Larger-context language modelling with recurrent neural network. In *Proceedings of ACL*, pages 1319–1329.

Warrington, E. K. (1975). The Selective Impairment of Semantic Memory. *Quarterly Journal of Experimental Psychology*, 27(4):635–657.

Warrington, E. K. and McCarthy, R. A. (1987). Categories of knowledge. *Brain*, 110(5):1273–1296.

Weber, N., Balasubramanian, N., and Chambers, N. (2018). Event representations with tensor-based compositions. In *Proceedings of AAAI*, pages 4946–4953.

Weeds, J., Clarke, D., Reffin, J., Weir, D., and Keller, B. (2014). Learning to distinguish hypernyms and co-hyponyms. In *COLING*, pages 2249–2259.

Weeds, J. and Weir, D. (2003). A general framework for distributional similarity. In *Proceedings of EMNLP*, pages 81–88.

Weeds, J., Weir, D., and McCarthy, D. (2004). Characterising measures of lexical distributional similarity. In *Proceedings of COLING*, pages 1015–1021.

Weston, J., Chopra, S., and Bordes, A. (2015). Memory Networks. *International Conference on Learning Representations*, pages 1–14.

Wieting, J., Bansal, M., Gimpel, K., and Livescu, K. (2015). From Paraphrase Database to Compositional Paraphrase Model and Back. *Transactions of the ACL*, 3:345–358.

Wu, Z. and Palmer, M. (1994). Verb semantics and lexical selection. In *Proceedings of ACL*, pages 133–138.

Yang, Y., Yuan, S., Cer, D., Kong, S.-Y., Constant, N., Pilar, P., Ge, H., Sung, Y.-h., Strope, B., and Kurzweil, R. (2018). Learning semantic textual similarity from conversations. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 164–174, Melbourne, Australia. Association for Computational Linguistics.

Yih, S. W.-T., Zweig, G., and Platt, J. C. (2012). Polarity inducing latent semantic analysis. In *Proceedings of EMNLP*, pages 1212–1222.

Yu, M. and Dredze, M. (2014). Improving lexical embeddings with semantic knowledge. In *Proceedings of ACL*, pages 545–550.

Zamani, H. and Croft, W. B. (2016). Embedding-based query language models. In *Proceedings of ICTIR*, pages 147–156.

Zaremba, W., Sutskever, I., and Vinyals, O. (2015). Recurrent neural network regularization. *Proceedings of ICLR*, pages 1–8.

Zeiler, M. D. (2012). ADADELTA: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *Proceedings of ECCV*, pages 818–833.

Zhu, Y., Vulić, I., and Korhonen, A. (2019). A systematic study of leveraging subword information for learning word representations. In *Proceedings of NAACL-HLT*.

Zipf, G. K. (1949). *Human behavior and the principle of least effort: An introduction to human ecology*.