Fully Delexicalized Contexts for Syntax-Based Word Embeddings

Jenna Kanerva
TurkuNLP Group
University of Turku
Graduate School (UTUGS)
Turku
Finland
jmnybl@utu.fi

Sampo Pyysalo
Language Technology Lab
DTAL
University of Cambridge
United Kingdom
sampo@pyysalo.net

Filip Ginter
TurkuNLP Group
University of Turku
Finland
figint@utu.fi

Abstract

Word embeddings induced from large amounts of unannotated text are a key resource for many NLP tasks. Several recent studies have proposed extensions of the basic distributional semantics approach where words form the context of other words, adding features from e.g. syntactic dependencies. In this study, we look in a different direction, exploring models that leave words out entirely, instead basing the context representation exclusively on syntactic and morphological features. Remarkably, we find that the resulting vectors still capture clear semantic aspects of words in addition to syntactic ones. We assess the properties of the vectors using both intrinsic and extrinsic evaluations, demonstrating in a multilingual parsing experiment using 55 treebanks that fully delexicalized syntax-based word representations give a higher average parsing performance than conventional word2vec embeddings.

1 Introduction

The recent resurgence of interest in neural methods for natural language processing involves a particular focus on neural approaches to inducing representations of words from large text corpora based on distributional semantics approaches (Bengio et al., 2003; Collobert et al., 2011). The methods introduced by Mikolov et al. (2013a) and implemented in their popular word2vec tool have been proven both effective and a good foundation for further exploration. In addition to representing word contexts as sliding windows of words in linear sequence, recent work has included efforts of building the word vectors using dependency-based approaches (Levy and Goldberg, 2014), where the context is based on nearby words in the syntactic tree.

In this paper, we set out to study dependency-based contexts further, exploring word embeddings derived from fully delexicalized syntactic contexts, and in particular the degree to which models induced using such context representations are dependent on word forms.

2 Methods

Our study builds on the seminal work introducing word2vec and later efforts generalizing it from a linear representation of context words to arbitrary contexts. We next present these methods and our proposed formulation of delexicalized syntax-based word embeddings.

2.1 Word2vec embeddings

The word2vec tool[1] implements two related approaches for inducing word representations – continuous bag-of-words (CBOW) and skip-grams – as well as a number of ways to train and parametrise them (Mikolov et al., 2013a; Mikolov et al., 2013b). Of these variants, the skip-gram with negative sampling (SGNS) model has been shown to be particularly effective and has become a *de facto* standard for neural word vector induction and the basis for many recent studies in the field. While the original work of Mikolov et al. explored different model architectures and approaches to learning, they all shared the property that the contexts of words in the model consisted of words.

2.2 Dependency-based word embeddings

Observing that the SGNS model is not inherently restricted to working with contexts consisting of words, Levy and Goldberg (2014) extended the model to work with arbitrary contexts, focusing

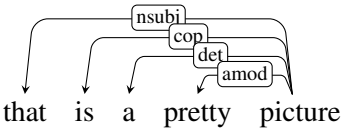
[1] https://code.google.com/p/word2vec/

Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017), pages 83-91,
Pisa, Italy, September 18-20 2017

in particular on dependency-based contexts consisting of combinations of a neigbouring word in the dependency graph and its dependency relation to the target word (e.g. `scientist/nsubj`). Compared to embeddings based on linear contexts of words, they showed dependency-based embeddings to emphasize functional over topical similarity and to have benefits in distinguishing word relatedness from similarity. Levy and Goldberg released their generalized version of `word2vec` allowing arbitrary contexts as `word2vecf`.[2]

2.3 Delexicalized syntax-based embeddings

Although the context definition of Levy and Goldberg incorporates dependency information, it remains lexicalized, including also the surface form of the dependent or head word. Here, we consider whether it is possible to induce useful word embeddings with *delexicalized* contexts that omit the word form entirely. Specifically, we define the context of a target word as 1) the set of all dependency relations headed by the target word, 2) the relation where the target word is the dependent, marked to differentiate it from those in set 1), 3) the part-of-speech tag of the target word, and 4) the set of morphological features assigned to the target word. This context definition is illustrated in Figure 1. We use the `word2vecf` implementation to create embeddings using this context definition.



word	context	word	context
that	PRON	a	PronType=Art
that	PronType=Rel	a	det
that	nsubj	pretty	ADJ
is	AUX	pretty	Degree=Pos
is	Mood=Ind	pretty	amod
is	Number=Sing	picture	NOUN
is	Person=3	picture	Number=Sing
is	Tense=Pres	picture	root
is	VerbForm=Fin	picture	Dep_nsubj
is	cop	picture	Dep_cop
a	DET	picture	Dep_det
a	Definite=Ind	picture	Dep_amod

Figure 1: Delexicalized context for words in an English sentence.

3 Experimental setup

We next present the sources of the unannotated texts and their syntactic analyses used as input and the methods and resources applied to create word embeddings and evaluate them.

3.1 Texts and dependency analyses

The texts used to induce word vectors are derived from the multilingual text collection recently introduced by Ginter et al. (2017) covering 45 languages. This resource consists primarily of texts collected through a combination of Internet crawl and extraction from Wikipedia data. The sizes of the 45 language-specific subcorpora range from 29,000 tokens for Old Church Slavonic to 9.5 billion tokens for English, averaging approximately 2B tokens with roughly half of the languages staying under the 1B token range. In addition to plain texts, the resource provides also full syntactic analyses following Universal Dependencies (UD) (Nivre et al., 2016) version 2.0 guidelines, including tokenization, lemmatization, full morphological analyses and parses produced with the UDPipe pipeline (Straka et al., 2016). We note that even though many languages in the UD collection are covered by more than one treebank (and analyses may differ across treebanks for a single language), only one set of automatic analyses are provided per language in this resource.

3.2 Embeddings

We use the `word2vec` embeddings provided together with the CoNLL 2017 Shared Task automatically analyzed corpora (Ginter et al., 2017) as a baseline in our experiments. These models are trained on tokenized and lowercased text using the SGNS approach with a window size of 10, minimum word frequency count 10, and 100-dimensional vectors. Our new delexicalized `word2vecf` embeddings are created using the same, identically tokenized and lowercased texts, where the UDPipe morphological and syntactic analyses are used to generate our syntax-based contexts. We use the same minimum word frequency count 10 and vector dimensionality of 100 for our `word2vecf` models.

[2] `https://bitbucket.org/yoavgo/word2vecf`

france	jesus	xbox	reddish	scratched	megabits
belgium	christ	playstation	brownish	knicked	megabit
luxembourg	jesus.	ps3	yellowish	bruised	kilobits
nantes	god	ps4	greenish	nicked	gigabits
marseille	ahnsahnghong	xbox360	pinkish	scuffed	mbps
bretagne	jesuschrist	wii	grayish	chewed	mbits
boulogne	y'shua	xbla	bluish	sandpapered	terabits
poitou	christ	psvita	-orange	scratches	mbit
rouen	christ.	titanfall	orangish	brusied	kbits
paris	jesus	xboxone	greyish	scraped	kilobit
toulouse	yeshua	gamecube	mid-brown	thwacked	megabytes

Table 1: Nearest neighbours in `word2vec` embeddings

3.3 Intrinsic evaluation

Word vectors are frequently evaluated by assessing how well their distance correlates with human judgments of word similarity. Although these intrinsic evaluations have known issues (see e.g. Batchkarov et al. (2016), Chiu et al. (2016), Faruqui et al. (2016)) and we agree with the criticism that they are frequently poor indicators of the merits of representations, we include this common form of intrinsic evaluation here for reference purposes. We provide results using a comprehensive collection of English datasets annotated for word similarity and relatedness. Specifically, we used the evaluation service introduced by Faruqui and Dyer (2014) to evaluate on the 13 datasets available on the service[3] at the time of this writing. The datasets are summarized below in Table 3.

3.4 Extrinsic evaluation

Our primary evaluation is based on dependency parsing, where we evaluate parsing accuracy using different pre-trained word embeddings during parser training. We use the UDPipe pipeline[4] for tokenizing, tagging, lemmatizing and parsing Universal Treebanks (Straka et al., 2016). In all experiments, we use system parameters optimized on baseline models separately for each treebank,[5] keeping the parameters fixed in the comparative evaluations of the different word representations. We note that any possible bias introduced by this parameter selection strategy would favour the baseline model rather than one using the delexicalized syntax-based representations proposed here.

Parsing results are reported for all UD v2.0 treebanks in the CoNLL 2017 Shared Task release[6] that have a separate development set which can be used for testing and raw data for training embeddings. Of the 64 treebanks in the release, 9 do not fulfill these criteria (French-ParTUT, Galician-TreeGal, Irish, Kazakh, Latin, Slovenian-SST, Ukrainian and Uyghur do not have development data, Gothic does not have raw data) and are not included in the evaluation. Models are trained on the training section of a treebank and tested on the development section.[7]

4 Results

We next informally illustrate the characteristics of the English word vectors using nearest neighbours and give the intrinsic evaluation results for these vectors before presenting the results of our primary multilingual parsing experiments.

4.1 Nearest neighbours

Table 1 shows nearest neighbours in the conventional `word2vec` embeddings using the cosine similarity metric for a somewhat arbitrary selection of English words.[8] As has been well established in previous work, near words in `word2vec` representations are commonly (near) synonyms (e.g. *jesus*/*christ*, *scratched*/*scuffed*), cohyponyms (*france*/*belgium*, *xbox*/*playstation*), or topically related (*france*/*paris*, *scratched*/*sandpaper*).

We expected that the use of delexicalized contexts would eliminate much of the ability of the

[3] `http://wordvectors.org/`

[4] `http://ufal.mff.cuni.cz/udpipe`

[5] Optimized UDPipe parameters for UD v2.0 treebanks are released in the supplementary data of UDPipe models at `http://hdl.handle.net/11234/1-1990`.

[6] `http://hdl.handle.net/11234/1-1983`

[7] The test sections of the treebanks were held out for the final shared task evaluation and were thus not available for our experiments.

[8] The choice of words follows a similar illustration by Collobert et al. (2011).

france	jesus	xbox	reddish	scratched	megabits
lebanon	osama	vbox	greenish	snatched	megabytes
australia	napoleon	whitesox	grayish	touched	microseconds
england	ophelia	matchbox	bluish	punched	hectares
bolivia	gautama	firefox	greyish	deflected	tonnes
scotland	scipio	wmp	pinkish	warmed	microns
estonia	sauron	audiovox	yellowish	levelled	micrograms
switzerland	chandragupta	virtualbox	brownish	booted	litres
finland	claudius	equinox	blackish	stalked	megawatts
slovenia	jamarcus	rotax	temperate	ditched	gallons
algeria	olivia	hmp	redish	swallowed	bushels

Table 2: Nearest neighbours in delexicalized syntax-based word embeddings

embeddings to organize words by factors such as synonymy, cohyponymy, and topic and that nearest neighbours in our delexicalized syntax-based representations would be associated much more loosely, by syntactic behaviour rather than any aspect of meaning. Of the words illustrated in Table 2, *scratched* and *xbox* can be seen as broadly following this expected pattern in neighbouring past form verbs and singular nouns (respectively) with little semantic coherence. However, by contrast, all ten words nearest to *france* are countries, the neighbours of *jesus* are first names, nine out of ten nearest to *reddish* have the form color*ish*, and *megabits* is nearest ten different units. This unexpected result suggests that the syntactic structures and morphological features associated with a word can generate surprisingly useful word representations even in the absence of any lexical information. We also note the concerning (and systematic) tendency for nearest neighbours to end with the same characters (e.g. 8/10 nearest *xbox* in *x*). Although this may seem very surprising, we ruled out the possibility of leaking any word-suffix information by obtaining the same results when only word hashes were used during the model training. Our explanation is to note that the effect is strongest for rare words and that the parses are generated with a complex statistical model with access to word surface forms which are indirectly reflected in the predicted morphological and syntactic structures. In particular, the POS and morphological tagger naturally uses word suffix information, and we hypothesize that the vector model is able to pick this weak signal from the output of the morphological tagger and syntactic parser.

4.2 Intrinsic evaluation results

The results for the intrinsic evaluation based on the comparison of word pair similarity ranking with human judgments on 13 datasets are summarized in Table 3. The correlations seen for the `word2vec` embeddings are in line with those for previously released representations generated using the algorithm (e.g. (Mikolov et al., 2013a)), confirming that the texts used to induce these representations are appropriate for generating high-quality word embeddings.

The results for the delexicalized syntax-based embeddings are, as expected, much lower and far from competitive on any of the datasets. Nevertheless, the correlations remain positive in all 13 evaluations, providing support for the proposition that delexicalized contexts representations can identify similarities in word meaning.

4.3 Dependency parsing results

Parsing performance for the 55 treebanks is summarized in Table 4. We report labeled attachment scores evaluated using gold standard word segmentation with predicted part-of-speech tags and morphological features for parsers trained using three different pre-trained word embeddings: `word2vec` embeddings trained on the texts of the manually annotated UD treebanks (baseline), `word2vec` embeddings trained on the large unannotated corpora, and our delexicalized syntax-based embeddings trained on the automatically analyzed corpora.

`word2vec` embeddings trained on the large unannotated corpora yield on average a +0.16% point improvement over the baseline model. Somewhat surprisingly, incorporating standard `word2vec` embeddings trained on the larger cor-

Dataset	Correlation word2vec	word2vecf	Pairs Found	Total	Reference
WordSim-353	0.7083	0.2350	353	353	Finkelstein et al. (2001)
WordSim-353-SIM	0.7677	0.4033	203	203	Agirre et al. (2009)
WordSim-353-REL	0.6691	0.1318	252	252	Agirre et al. (2009)
MC-30	0.7028	0.2929	30	30	Miller and Charles (1991)
RG-65	0.6801	0.0593	65	65	Rubenstein and Goodenough (1965)
Rare-Word	0.4250	0.1998	2006	2034	Luong et al. (2013)
MEN	0.7397	0.2027	3000	3000	Bruni et al. (2012)
MTurk-287	0.6958	0.3474	287	287	Radinsky et al. (2011)
MTurk-771	0.6406	0.1336	771	771	Halawi et al. (2012)
YP-130	0.3882	0.0464	130	130	Yang and Powers (2006)
SimLex-999	0.3376	0.1004	999	999	Hill et al. (2016)
Verb-143	0.3633	0.2425	144	143	Baker et al. (2014)
SimVerb-3500	0.2175	0.0476	3500	3500	Gerz et al. (2016)

Table 3: Intrinsic evaluation results. The numbers of found pairs are identical for the two methods.

pora produces notably worse results compared to the baseline model for a number of languages. For Old Church Slavonic, the over 2% point drop in performance can likely be attributed to the modest size of the unannotated corpus available for that language: only 29,000 words are available in the raw data collection, compared to 37,500 words in the treebank training set. Otherwise, the differences range between -1.55% points and +6.28% points, with 31 treebanks showing positive results and 23 negative results. While some of these negative effects may be attributable to domain mismatches between the treebanks and the web-crawled and Wikipedia-derived texts, further study is required to analyze these findings in detail.

The delexicalized syntax-based embeddings yield an average 0.88% point improvement. Excluding Old Church Slavonic, which behaves similarly as with `word2vec` embeddings, the difference to the baseline ranges between -0.80% points and +7.30% points, with 45 treebanks showing a positive effect and 9 negative results. Overall, our results indicate the surprising conclusion that delexicalized syntactic embeddings lead to higher performance than conventional `word2vec` embeddings as well as generalize better across languages when evaluated in this closely related task.

4.4 Analysis

Given the positive effects of delexicalized syntax-based embeddings on the parsing task, it is natural to ask how the baseline parser performance affects the quality of the word embeddings. We set out to test this on Finnish, where our syntax-based embeddings have a clear positive effect compared to conventional `word2vec` embeddings and where our baseline parser accuracy is relatively low compared to the state-of-the-art parsers.

We first study whether the better parsing model showing a 1.65% point improvement in labeled attachment score can be used in a bootstrapping setup to generate yet better embeddings and parsers. We parsed the Finnish raw data with this better model, induced word vectors on the newly parsed data, and trained a UDPipe parsing model with the newly created word vectors. The results of this experiment are shown in Table 5. In terms of LAS, the second iteration model is +0.23% points better than the model from the first iteration.

We note that UDPipe may not be the optimal parsing pipeline for this experiment: our syntax-based embeddings are trained using both morphological features and syntactic trees, but while the UDPipe parser (Parsito (Straka et al., 2015)) uses pre-trained embeddings, the morphological tagger (MorphoDiTa (Straková et al., 2014)) does not, thus leaving part-of-speech tags and morphological features intact in newly parsed data. This means that the difference between old and new vector training data is relatively small.

A second consideration is that the 75.7% accuracy of the baseline parser used is not competitive with state-of-the-art parsers, where best reported labeled attachment scores for Finnish are in the range of 83-84% (Alberti et al., 2017; Bohnet et al., 2013). To investigate the effect of using higher-quality parses, we trained our syntax-based embeddings on the Finnish Internet Parsebank (Luotolahti et al., 2015), a 3.6 billion token collection of web crawled data. Finnish Internet Parsebank is analyzed with the Finnish de-

language	baseline	word2vec	diff to baseline	syntax-based	diff to baseline
Ancient_Greek	56.61	57.93	+1.32	58.18	+1.57
Ancient_Greek-PROIEL	72.35	72.48	+0.13	72.67	+0.32
Arabic	72.88	73.91	+1.03	74.00	+1.12
Basque	69.02	69.74	+0.72	69.93	+0.91
Bulgarian	83.90	84.29	+0.39	85.18	+1.28
Catalan	85.15	85.01	-0.14	85.31	+0.16
Chinese	68.48	68.83	+0.35	69.06	+0.58
Croatian	76.08	75.98	-0.10	77.35	+1.27
Czech-CAC	83.75	83.58	-0.17	84.54	+0.79
Czech-CLTT	69.58	68.92	-0.66	72.19	+2.61
Czech	84.47	84.24	-0.23	84.69	+0.22
Danish	75.18	74.63	-0.55	74.99	-0.19
Dutch-LassySmall	75.67	75.01	-0.66	76.68	+1.01
Dutch	74.73	75.21	+0.48	75.00	+0.27
English	79.66	80.20	+0.54	80.64	+0.98
English-LinES	74.62	74.35	-0.27	75.59	+0.97
English-ParTUT	75.72	75.21	-0.51	76.20	+0.48
Estonian	60.65	61.89	+1.24	63.22	+2.57
Finnish	75.70	75.79	+0.09	77.35	+1.65
Finnish-FTB	76.42	76.68	+0.26	77.72	+1.30
French	86.08	85.71	-0.37	86.53	+0.45
French-Sequoia	82.30	82.58	+0.28	82.65	+0.35
Galician	77.58	77.34	-0.24	78.21	+0.63
German	73.10	73.12	+0.02	72.87	-0.23
Greek	79.04	77.93	-1.11	79.93	+0.89
Hebrew	76.88	77.38	+0.50	78.52	+1.64
Hindi	87.09	86.82	-0.27	87.38	+0.29
Hungarian	65.59	66.40	+0.81	68.44	+2.85
Indonesian	74.39	72.84	-1.55	73.59	-0.80
Italian	85.44	84.98	-0.46	84.96	-0.48
Italian-ParTUT	78.21	78.74	+0.53	79.92	+1.71
Japanese	93.09	93.09	+0.00	93.23	+0.14
Korean	56.42	62.70	+6.28	63.72	+7.30
Latin-ITTB	71.15	71.72	+0.57	72.98	+1.83
Latin-PROIEL	70.08	69.76	-0.32	69.89	-0.19
Latvian	64.01	64.56	+0.55	66.16	+2.15
Norwegian-Bokmaal	83.91	83.44	-0.47	84.18	+0.27
Norwegian-Nynorsk	82.32	81.65	-0.67	81.89	-0.43
Old_Church_Slavonic	73.56	71.22	-2.34	71.40	-2.16
Persian	80.38	79.56	-0.82	80.86	+0.48
Polish	79.42	80.62	+1.20	81.21	+1.79
Portuguese-BR	85.55	86.11	+0.56	86.26	+0.71
Portuguese	83.64	84.49	+0.85	84.93	+1.29
Romanian	79.82	79.77	-0.05	80.30	+0.48
Russian	75.41	76.00	+0.59	77.48	+2.07
Russian-SynTagRus	86.76	86.58	-0.18	87.71	+0.95
Slovak	75.39	75.65	+0.26	76.55	+1.16
Slovenian	80.62	80.87	+0.25	81.38	+0.76
Spanish-AnCora	84.17	84.55	+0.38	84.31	+0.14
Spanish	84.34	83.85	-0.49	84.11	-0.23
Swedish-LinES	74.35	74.72	+0.37	75.34	+0.99
Swedish	73.39	74.25	+0.86	74.75	+1.36
Turkish	56.00	56.24	+0.24	57.75	+1.75
Urdu	76.98	76.23	-0.75	76.26	-0.72
Vietnamese	55.85	56.26	+0.41	55.22	-0.63
Average	-	-	+0.16	-	+0.88

Table 4: Parsing results for Conll 2017 shared task UD treebanks using different pretrained word embeddings. Green colour identifies treebanks where the performance of delexicalized syntax-based embeddings is higher than standard `word2vec` embeddings and the difference to the baseline model is positive.

	baseline	iteration 1	iteration 2
Finnish	75.70	77.35	77.57

Table 5: Bootstrapping results for Finnish syntax-based embeddings.

pendency parsing pipeline[9] trained on the UD Finnish treebank (Pyysalo et al., 2015) version 1.2. The Finnish parsing pipeline uses the OMorFi rule-based morphological analyzer (Pirinen, 2008) converted to the UD scheme, the Marmot tagger (Müller et al., 2013) and the graph-based dependency parser of Bohnet (2010). The labeled attachment score of the pipeline is estimated to be 82% based on the experiments reported in Pyysalo et al. (2015).

Interestingly, when the UDPipe parser was trained with syntax-based word embeddings induced from Finnish Internet Parsebank, UDPipe performance improved to the general level of the original parser used, giving a LAS of 82.21%. It must be noted that this number is not comparable to our main parsing results as the version of the UD Finnish treebank is different (version 1.2 compared to version 2.0), and the raw text collection is more than three times bigger. With UDPipe using standard `word2vec` pre-trained embeddings trained on the same Finnish Internet Parsebank data, parsing accuracy was 78.35%. These preliminary results are very promising and indicate that with good pre-trained word embeddings, we are able to improve a fast and comparatively simple feedforward parser near the numbers of the new DRAGNN-based SyntaxNet (Kong et al., 2017; Alberti et al., 2017) parser, which is more complex and much slower. Currently, we were only able to "mimic" the numbers of a good parser as we needed a high-quality parsebank to achieve these results, and the question whether similar results could be obtained without the near state-of-the-art parser remains open.

5 Conclusions and Future Work

In this work, we proposed a fully delexicalized syntax-based context representation for inducing word vectors using the Levy and Goldberg (2014) generalization of the `word2vec` skip-gram with negative sampling (SGNS) model. Building on a recently developed large-scale multilingual resource of texts automatically annotated with Universal Dependencies, we created delexicalized syntax-based word embeddings for 45 different languages. Examination of nearest neighbours and evaluation against 13 English datasets annotated for human judgments of word similarity suggested that the embeddings retained a substantial degree of information on not only the syntactic and morphological aspects of words but also on aspects of their meaning despite being induced through a process with no access to lexical information. An extensive extrinsic evaluation using the UDPipe parser and 55 CoNLL 2017 shared task corpora demonstrated that the addition of our syntax-based embeddings not only substantially improved the performance of the baseline UDPipe model on average, but also that this improvement was greater than when using standard `word2vec` SGNS embeddings. A detailed analysis on Finnish showed potential additional promise from approaches using bootstrapping as well as combinations of embeddings induced using parses generated using complex models in simpler and faster parsers.

Our initial exploration suggests that fully delexicalized syntax-based embeddings have intriguing properties and show promise for use in practical applications. In future work, we will further explore how delexicalized context representations can capture aspects of word meaning – both in terms of degree and mechanism – as well as explore their use in improving mono- and multilingual parsing performance in combination with state-of-the-art models.

[9] `https://github.com/TurkuNLP/Finnish-dep-parser`

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of HLT-NAACL'09*, pages 19–27.

Chris Alberti, Daniel Andor, Ivan Bogatyy, Michael Collins, Dan Gillick, Lingpeng Kong, Terry Koo, Ji Ma, Mark Omernick, Slav Petrov, et al. 2017. Syntaxnet models for the conll 2017 shared task. *arXiv preprint arXiv:1703.04929*.

Simon Baker, Roi Reichart, and Anna Korhonen. 2014. An unsupervised model for instance level subcategorization acquisition. In *Proceedings of EMNLP'14*, pages 278–289.

Miroslav Batchkarov, Thomas Kober, Jeremy Reffin, Julie Weeds, and David Weir. 2016. A critique of

word similarity as a method for evaluating distributional semantic models. In *Proceedings of RepEval'16*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Bernd Bohnet, Joakim Nivre, Igor Boguslavsky, Richrd Farkas, Filip Ginter, and Jan Haji. 2013. Joint morphological and syntactic analysis for richly inflected languages. *Transactions of the Association for Computational Linguistics*, 1:415–428.

Bernd Bohnet. 2010. Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd international conference on computational linguistics*, pages 89–97.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of ACL'12*, pages 136–145.

Billy Chiu, Anna Korhonen, and Sampo Pyysalo. 2016. Intrinsic evaluation of word vectors fails to predict extrinsic performance. In *Proceedings of RepEval'16*, pages 1–6.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Manaal Faruqui and Chris Dyer. 2014. Community evaluation and exchange of word vectors at wordvectors.org. In *Proceedings of ACL'14*.

Manaal Faruqui, Yulia Tsvetkov, Pushpendre Rastogi, and Chris Dyer. 2016. Problems with evaluation of word embeddings using word similarity tasks. In *Proceedings of RepEval'16*, pages 30–35.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of WWW'01*, pages 406–414.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. SimVerb-3500: A large-scale evaluation set of verb similarity. In *Proceedings of EMNLP'16*.

Filip Ginter, Jan Hajič, Juhani Luotolahti, Milan Straka, and Daniel Zeman. 2017. CoNLL 2017 shared task - automatically annotated raw texts and word embeddings. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Guy Halawi, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. 2012. Large-scale learning of word relatedness with constraints. In *Proceedings of SIGKDD'12*, pages 1406–1414. ACM.

Felix Hill, Roi Reichart, and Anna Korhonen. 2016. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Lingpeng Kong, Chris Alberti, Daniel Andor, Ivan Bogatyy, and David Weiss. 2017. Dragnn: A transition-based framework for dynamically connected neural networks. *arXiv preprint arXiv:1703.04474*.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of ACL*.

Thang Luong, Richard Socher, and Christopher D Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of CoNLL'13*, pages 104–113.

Juhani Luotolahti, Jenna Kanerva, Veronika Laippala, Sampo Pyysalo, and Filip Ginter. 2015. Towards universal web parsebanks. In *Proceedings of the International Conference on Dependency Linguistics (Depling'15)*, pages 211–220. Uppsala University.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of ICLR*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, pages 3111–3119.

George A Miller and Walter G Charles. 1991. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

Thomas Müller, Helmut Schmid, and Hinrich Schütze. 2013. Efficient higher-order crfs for morphological tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666.

Tommi Pirinen. 2008. Suomen kielen äärellistilainen automaattinen morfologinen jäsennin avoimen lähdekoodin resurssein. *University of Helsinki*.

Sampo Pyysalo, Jenna Kanerva, Anna Missilä, Veronika Laippala, and Filip Ginter. 2015. Universal dependencies for Finnish. In *Proceedings of NODALIDA 2015, May 11-13, 2015, Vilnius, Lithuania*, pages 163–172.

Kira Radinsky, Eugene Agichtein, Evgeniy Gabrilovich, and Shaul Markovitch. 2011. A word at a time: computing word relatedness using

temporal semantic analysis. In *Proceedings of WWW'11*, pages 337–346.

Herbert Rubenstein and John B Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Milan Straka, Jan Hajič, Jana Straková, and Jan Hajič jr. 2015. Parsing universal dependency treebanks using neural networks and search-based oracle. In *Proceedings of Fourteenth International Workshop on Treebanks and Linguistic Theories (TLT 14)*, December.

Milan Straka, Jan Hajič, and Jana Straková. 2016. UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of LREC'16*.

Jana Straková, Milan Straka, and Jan Hajič. 2014. Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 13–18, Baltimore, Maryland, June. Association for Computational Linguistics.

Dongqiang Yang and David MW Powers. 2006. Verb similarity on the taxonomy of wordnet. In *Proceedings of GWC'06*.