Published in: Bański, Piotr; Biber, Hanno; Breiteneder, Evelyn; Kupietz, Marc; Lüngen, Harald; Witt, Andreas (eds.) (2015): Proceedings of the 3rd Workshop on Challenges in the Management of Large Corpora (CMLC-3). Mannheim: Institut für Deutsche Sprache, pp. 11-14.

Discovering Subtle Word Relations in Large German Corpora

Sebastian Buschjäger
Lehrstuhl für Künstliche Intelligenz
TU Dortmund

Lukas Pfahler
Lehrstuhl für Künstliche Intelligenz
TU Dortmund

Katharina Morik
Lehrstuhl für Künstliche Intelligenz
TU Dortmund

{sebastian.buschjaeger,lukas.pfahler,katharina.morik}@udo.edu

Abstract

With an increasing amount of text data available it is possible to automatically extract a variety of information about language. One way to obtain knowledge about subtle relations and analogies between words is to observe words which are used in the same context. Recently, Mikolov et al. proposed a method to efficiently compute Euclidean word representations which seem to capture subtle relations and analogies between words in the English language. We demonstrate that this method also captures analogies in the German language. Furthermore, we show that we can transfer information extracted from large non-annotated corpora into small annotated corpora, which are then, in turn, used for training NLP systems.

1 Motivation

Large text corpora are a rich source of information for testing language properties. Once we formulate a linguistic hypothethis, we can formulate queries to collect evidence from the corpus (Klein and Geyken, 2010). However, very large corpora allow us to perform automatic exploration of the corpus to identify subtle relations between words or word groups.

Unfortunately, the analysis of large corpora is computationally challenging. As the size of a corpus grows, the size of the used vocabulary also grows, because a larger subset of language is covered. We found that the German Wikipedia contains more than 1.6 million unique words.

In order to find instances of all possible word-word relations or word classes, a very large sample of text data must be drawn. We usually refer to this problem as the “curse of dimensionality”. However, for most Natural Language Problems, only little annotated training data is available.

Recently, Mikolov et al. (2013c) introduced a method for discovering linguistic regularities in large corpora based on neural networks. Their method learns a mapping from words to vectors in $\mathbb{R}^D$ called *word embeddings*. Embeddings allow simple vector operations that seem to capture syntactical and semantical regularities. This method has been successfully applied to English text corpora. For the first time, we thoroughly evaluate this method for the German language.

Our goal is to extract information on word relations from large unannotated corpora and enrich smaller annotated corpora like the TüBa-D/Z treebank (Telljohann et al., 2009) – a collection of German newspaper articles – with this information. More specifically, we want to discover word similarities and analogies in order to aggregate words into groups.

The rest of this paper is organized as follows. In section 2 we formally introduce Mikolov’s word embeddings, in section 3 we present our experiments for German and English Wikipedia documents. Then in section 4 we show related work. Section 5 concludes our work.

2 Word Embeddings

Mikolov et al. proposed a neural language model that estimates word transition probabilities from a training corpus (2013c). By gradually reducing the complexity of their model, the authors enable the efficient use of large text corpora resulting in a simple neural network with input layer, linear projection layer and log-linear output layer (Mikolov et al., 2013a; Mikolov et al., 2013b). The projection layer of this model implicitly calculates a mapping $u : \mathcal{V} \mapsto \mathbb{R}^D$ from the vocabulary $\mathcal{V}$ to the space of word embeddings $\mathbb{R}^D$.

Surprisingly, these embeddings show striking syntactic and semantic properties that allow us to perform simple vector operations, e.g.,

$$u(Paris) - u(France) + u(Italy) \approx u(Rome)$$

In order to train such an embedding, Mikolov et al. present two closely related network topologies (cf. figure 1). The first model, called CBOW, estimates probabilities for words $v_i \in \mathcal{V}$ given their surroundings $w_1, \ldots, w_N$ using a softmax function. Let U be a weight matrix shared across all contextual words $w_1, \ldots, w_N$ and let $W_{i\boldsymbol{.}}$ denote the $i-$th row of the output matrix W, then this model can be formulated as follows:

$$\begin{aligned} \tilde{u} &= \sum_{i=1}^{N} U w_i \\ p(v_j | w_1, \ldots, w_N) &= \frac{\exp(W'_{j\boldsymbol{.}} \tilde{u})}{\sum_{i=1}^{V} \exp(W'_{i\boldsymbol{.}} \tilde{u})} \end{aligned}$$

The second model, called Skip-Gram (SG), reverses the CBOW task. Given a single word $v_i \in \mathcal{V}$ it estimates the probabilities for the surrounding contextual words $w_1, \ldots, w_N$. The mathematical formulation for this model is naturally extracted from the CBOW model by adding multiple output matrices $W^{(1)}, \ldots, W^{(N)}$ to the model while reducing the input layer to one word.

The authors show, that the word embeddings u capture semantic relations between words by using simple vector operations. Additionally, they find that similar words have similar embeddings by the means of Cosine similarity. This enables efficient queries for word similarities in a vocabulary since the word embeddings can be efficiently computed as a look-up in table u and the Cosine similarity can be implemented as linear-time vector operation.

3 Experiments

3.1 Training German Word Embeddings

We train our word embeddings using the German Wikipedia (Wikimedia, 2015). This set contains roughly 591 million words with a vocabulary of 1.6 million words. As a comparison, word embeddings for the English Wikipedia with approximately 1.7 billion words and a vocabulary size of 1.7 million words are trained as well (Wikimedia, 2015). An available subset of word embeddings computed by Mikolov et al. on a large Google-News text corpus will serve as a reference value for our experiments (Mikolov, 2015).

3.2 Identifying Word Analogies

Mikolov et al. analyze the accuracy of word embeddings on semantic and syntactic relations based on a test set. This test set contains phrases of the form "a is to b what c is to d." for different categories of relations, e.g.

`king` is to `queen` what `man` is to `woman`

The task of this test set is to predict the word d where words a, b, c are given. To do so, a simple nearest neighbor prediction is used:

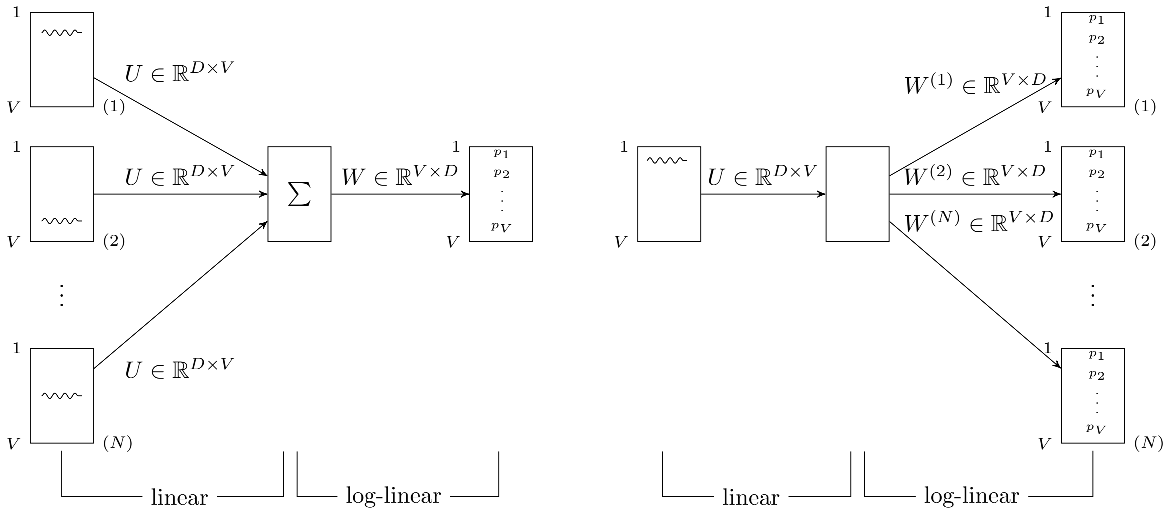
$$\widehat{d} = \underset{v \in \mathcal{V}}{\operatorname{argmin}} \{ \|u(a) - u(b) + u(c) - u(v)\|_2^2 \}$$

A question is correctly answered if $\widehat{d}$ equals d.

For the first time, we analyzed the accuracy of word embeddings in the German language. Therefore, we half-automatically translated this English test set into German using (Moraes, 2015). Additionally to this regularity test, we analyzed the performance of word embeddings on word analogies. To do so, we assembled a list of one thousand nouns for the German and English language. For every German noun, we queried twelve synonyms on average using OpenThesaurus (Naber, 2015). For the English language, OpenOffice (Foundation, 2015) provided a synonyms dictionary with thirteen synonyms per noun on average. We then computed the average Cosine similarity between word embeddings and their synonyms embeddings. As a reference we computed the average Cosine similarity between random nouns.

Results for the regularity test are presented in table 1. As you can see, the word embeddings capture regularities between nouns in the German language quite well (cf. category "capital-common" and "capital-world"), but show relatively poor performance on plural forms and past tense (cf. category "gram7" and "gram8"). Reasons for this may lie in the lexical character of the underlying training corpus, the relatively small size of the German Wikipedia compared to the English Wikipedia and Google News-Corpus as well as irregularities in word construction in the German language.

In table 2 the results of the synonym test can be found. The picture reverse here in contrast to the results in table 1. The average Cosine similarity for analogous words in the German language are roughly twice as high as for the English language. The average Cosine similarity between random nouns is, as expected, nearly zero.


1
V
$U \in \mathbb{R}^{D\times V}$
(1)
$U \in \mathbb{R}^{D\times V}$
(2)
$U \in \mathbb{R}^{D\times V}$
(N)
Σ
$W \in \mathbb{R}^{V\times D}$
p_1
p_2
p_V
linear
log-linear
$U \in \mathbb{R}^{D\times V}$
$W^{(1)} \in \mathbb{R}^{V\times D}$
$W^{(2)} \in \mathbb{R}^{V\times D}$
$W^{(N)} \in \mathbb{R}^{V\times D}$
(1)
(2)
(N)
linear
log-linear

(a) CBOW model

(b) Skip-Gram model

Figure 1: Network topology for CBOW and Skip-Gram model.

category	ref.	English	German	
		CBOW	SG	CBOW
capital-common	81.60	86.96 (+5.36)	**93.48** (+9.88)	91.70 (+8.1)
capital-world	83.30	**91.29** (+7.99)	82.55 (+1.25)	84.88 (+1.58)
gram7-past-tense	64.49	**65.26** (+0.8)	42.11 (-22.38)	42.17 (-22.32)
gram8-plural	86.64	**84.01** (+2.63)	45.16 (-41.48)	48.02 (-38.62)
gram9-pl-verb	67.93	62.07 (-5.86)	62.83 (-5.1)	**65.15** (-2.78)

Table 1: Accuracy for regularity test (excerpt).

	ref.	English	German	
		CBOW	SG	CBOW
synonyms	0.25	0.26	0.56	0.56
random nouns	0.08	0.04	0.06	0.05

Table 2: Average Cosine similarity.

3.3 Enriching Small Annotated Corpora with Word Embeddings

We want to demonstrate that natural language processing problems that rely on relatively small annotated corpora as training data can benefit from word embeddings learned on large, non-annotated corpora. We have seen that similar words have similar word embeddings. Clustering the embeddings with k-Means thus yields k partitions of similar words. Enriching a small annotated training corpus by tagging each word with the partition it belongs to has two possible advantages: First, we can handle unknown words the same way as words with similar embeddings. Second, we can pool related words and can estimate more reliable statistics for rare words (Andreas and Klein, 2014).

In our experiment, we consider the TüBa-D/Z treebank (Telljohann et al., 2009), a corpus of merely 3,444 newspaper articles whose sentences are annotated with dependence trees. This treebank is widely used for training natural language parsers for both constituency and dependency grammars. We evaluate a classification problem closely related to dependency parsing, where for an unlabeled arc in a given parsetree we want to predict the label of the arc. The TüBa-D/Z treebank in `.conll` dependency tree format has 34 classes of dependencies (Foth, 2006). We use Naive Bayes for classification using features for the word, the lemma and the POS-tag of both the head and tail of the arc. Additionally, we use the cluster of the word embedding for the corresponding word as a feature.

We select $k \approx \sqrt{1.6M}$, such that the space of pairs of words is about the size of the vocabulary. This makes estimating statistics about pairs of words feasible. Using a 10-fold, linearly split cross validation we show an accuracy of $87.33 \pm 0.43\%$ using only traditional features. Using the additional features based on word embedding clusters, we get an accuracy of $88.33 \pm 0.43\%$, which is a significant increase of 1%.

4 Related Work

There have been many attempts to incorporate word embeddings into existing natural language processing solutions for the English language. Examples include Named-Entity Recognition (Turian et al., 2009), Machine Translation (Zou et al., 2013), Sentiment Analysis (Maas et al., 2011) or Automatic Summarization (Kageback et al., 2014). For Natural Language Parsing, there have been attempts to improve parser training by incorporating new features based on word embeddings. Andreas and Klein investigated their usefulness for constituency parsing (2014), Hisamoto et al. (2013) and Bansal et al. (2014) for dependency parsing. Their features are also based on clustered word embeddings and they also report small, but significant increases in accuracy for English dependency parsing.

5 Conclusion

We have shown that word embeddings can capture word similarities and word analogies for the German language. We demonstrated a significant improvement of parse tree labeling accuracy for German TüBa-D/Z treebank based on word embeddings.

References

Jacob Andreas and Dan Klein. 2014. How much do word embeddings encode about syntax. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Kilian Foth. 2006. Eine Umfassende Dependenzgrammatik des Deutschen.

Apache Foundation. 2015. Lingucomponent Sub-Project: Thesaurus Development. `http://www.openoffice.org/lingucomponent/thesaurus.html`. [Online; accessed on 02/18/2015].

Sorami Hisamoto, Kevin Duh, and Yuji Matsumoto. 2013. An Empirical Investigation of Word Representations. In *Proceedings of ANLP*, number C.

Mikael Kageback, Olof Mogren, Nina Tahmasebi, and Devdatt Dubhashi. 2014. Extractive Summarization using Continuous Vector Space Models. *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)@EACL 2014*, pages 31–39.

Wolfgang Klein and Alexander Geyken. 2010. Das digitale Wörterbuch der deutschen Sprache (dwds). *Lexicographica*, 26:79–93.

Andrew L Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning Word Vectors for Sentiment Analysis. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013c. Linguistic regularities in continuous space word representations. In *2013 Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings*, pages 746–751.

Tomas Mikolov. 2015. word2vec – Tool for computing continuous distributed representations of words. `http://code.google.com/p/word2vec/`. [Online; accessed on 02/16/2015].

Manuela Moraes. 2015. Glosbe API. `https://glosbe.com/a-api`. [Online; accessed on 02/18/2015].

Daniel Naber. 2015. OpenThesaurus.de. `https://www.openthesaurus.de/about/download`. [Online; accessed on 02/18/2015].

Heike Telljohann, Erhard W Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. 2009. Stylebook for the Tübingen treebank of written German (TüBa-D/Z).

Joseph Turian, L Ratinov, Y Bengio, and D Roth. 2009. A preliminary evaluation of word representations for named-entity recognition. *NIPS Workshop on Grammar Induction*, pages 1–8.

Wikimedia. 2015. Wikipedia Dumps. `http://dumps.wikimedia.org/`. [Online; accessed on 02/24/2015].

Will Y Zou, Richard Socher, Daniel Cer, and Christopher D Manning. 2013. Bilingual Word Embeddings for Phrase-Based Machine Translation. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*.