Neural Networks Revisited for Proper Name Retrieval from Diachronic Documents

Irina Illina, Dominique Fohr

MultiSpeech team
Université de Lorraine, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France
Inria, Villers-lès-Nancy, F-54600, France
CNRS, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France

Abstract

Developing high-quality transcription systems for very large vocabulary corpora is a challenging task. Proper names are usually key to understanding the information contained in a document. To increase the vocabulary coverage, a huge amount of text data should be used. In this paper, we extend the previously proposed neural networks for word embedding models: word vector representation proposed by Mikolov is enriched by an additional ***non-linear transformation***. This model allows to better take into account lexical and semantic word relationships. In the context of broadcast news transcription and in terms of recall, experimental results show a good ability of the proposed model to select new relevant proper names.

Keywords: speech recognition, neural networks, vocabulary extension, out-of-vocabulary words, proper names

1. Introduction

In the context of *Large-Vocabulary Continuous Speech Recognition* (LVCSR) systems, accurate recognition of *proper names* (PNs) is important because proper names are essential for understanding the content of the speech (for example, for voice search, spoken dialog systems, broadcast news transcription, etc.). No vocabulary can contain all existing PNs (Friburger and Maurel, 2002). By searching new proper names and by adding them to the standard vocabulary of LVCSR, we want to face the problem of *out-of-vocabulary* words (OOV, words that are not in LVCSR system vocabulary).

In word similarity or analogy tasks, count-based distribution models (Turney and Pantel, 2010), (Baroni and Lenci, 2010), (Church and Hanks, 1990) and word embedding models have been successfully used (Bengio *et al*., 2015), (Deng *et al*. 2013). These approaches are based on the idea that words in similar contexts have similar meanings.

Recently, several new word embedding approaches have been proposed and have given very good performance. Mikolov *et al*. (2013a), (2013b), (2013c) have proposed continuous word representations in vector space based on Neural Networks (NN): semantic and syntactic word relationships are taken into account using huge amounts of unstructured text data. Another popular approach is *GloVe* (Global Vectors) that is based on a log-bilinear regression and tries to keep meaningful structure of the word space (Pennington *et al.*, 2014). Hyperparameter optimization is a crucial factor for performance gain for embedding systems (Levy *et al*., 2015a).

Compared to other methods, Mikolov's word embedding model gives very good accuracy on different tasks while minimizing computational complexity (Levy *et al*., 2015a). Today, the Mikolov's system represents a state-of-the-art framework.

In (Fohr and Illina, 2015) Mikolov's word embedding methods have been proposed for increasing the vocabulary of the ASR system with new PNs. The system uses lexical and temporal features. PNs evolve through time, and that for a given date, the same PNs would occur in documents that belong to the same time period (Federico and Bertoldi, 2001). In the present paper, the same problem of PN retrieval using lexical and temporal context is considered. We extend this work and propose to better model the word dependencies in Mikolov's word embedding model.

The scientific contributions of this paper are:

- We extend the Mikolov's neural network by adding an extra non-linear transformation and we study different word projections;
- We present a comparison of standard (Mikolov) and our proposed approach in the context of French broadcast news speech transcription.

The paper is organized as follows. Section 2 introduces the proposed approach. Sections 3 and 4 describe the experimental sets and the results of the evaluation of this model.

2. Proposed methodology

In this paper, we use the same general framework that in (Fohr and Illina, 2015), (Illina *et al.,* 2014): we want to use the relationships between co-occurring PNs for better OOV PN retrieval. For this, we want to take into account temporal, lexical and semantic context of PNs. We use text documents from a diachronic corpus that are contemporaneous with test documents to be transcribed. We assume that, for a certain date, a PN from the test corpus will co-occur with other PNs in diachronic documents of the same time period (Kobayashi *et al.*, 1998). So, we have a test audio document to transcribe which contains OOV words, and we have a diachronic text corpus used to retrieve OOV proper names. An augmented vocabulary is dynamically built for each test document to avoid an excessive increase of vocabulary size.

We chose the high-quality vector representation of words proposed by Mikolov *et al.* (2013b) for OOV PN retrieval. This approach allows to build semantic context dependencies of OOV PNs.

2.1. OOV retrieval method

Our OOV retrieval method consists of 5 steps as in (Fohr and Illina, 2015):

A) In-vocabulary (IV) PN extraction from each test document: For each test document, we extract IV PNs from the automatic

transcription performed using our standard vocabulary. The goal is to use these PNs as anchors to collect linked new proper names.

B) Selection of diachronic documents and extraction of new PNs from them: only diachronic documents (DDs) that correspond to the same time period as the test document are considered. After POS-tagging of these DDs, meaningful words are kept: verbs, adjectives, nouns and PNs. Among these PNs, we create a list of those that do not belong to our standard vocabulary (OOV PN).

C) Temporal and lexical context extraction from diachronic documents (DD): After extracting the list of the IV PNs from the test document (step A), and the list of the OOV PNs from DDs (step B), we build their *temporal and lexical* contexts. For this, a high-dimensionality word representation space is used (see description below). We hope that in this space semantically and lexically related words will be in the same region of the space.

D) Ranking of new PNs: The cosine-similarity metric is calculated between the projected vector of IV PNs found in the test document and the projected vector of each OOV PN occurring in the selected diachronic documents.

E) Vocabulary augmentation: To reduce the vocabulary growth, only the top-N OOV PNs according to the cosine-similarity metric are added to our vocabulary. OOV PN pronunciations are generated using a phonetic dictionary or an automatic phonetic transcription tool.

Using this methodology, we expect to extract a reduced list of the potentially missing PNs.

Figure 1 presents an example of the cosine-similarity computation for one OOV PN (step D).

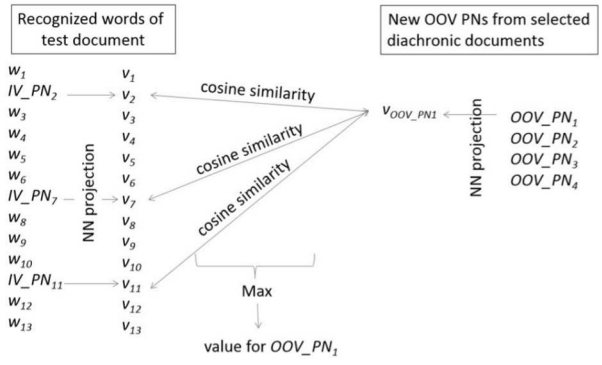

Figure 1: Example of computation of the cosine-similarity metric of an OOV_PN.

2.2. Neural networks for word representation

We propose to model the word space (step C) using Mikolov's neural network. In this network, each word is represented by a continuous vector in a high-dimensionality space. We hope that this space takes into account semantic and lexical relationships between words.

We use Mikolov's *Skip-gram* model that tries to predict surrounding words of one input word. This is performed by maximizing the classification rate of nearby words given the input word. More formally, given a sequence of training words $w_1, w_2 \ldots, w_T$, it maximizes the average *log* probability:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t) \qquad (1)$$

where *c* is the context size and T the number of training words. Compared to classical NN, the non-linear hidden layer is removed and the projection layer is shared for all words. This model assumes that semantically similar words will be projected in the same region of the word semantic space. An important property of this model is that the word representations learned by the *Skip-gram* model exhibit a linear structure.

Standard case (Mikolov's model)

Let $X=\{x_1, \ldots, x_V\}$ denotes the input vector of the neural network. For a given input word, only one element of X will be *1*, all other elements will be *0*. V is the vocabulary size and N is the size of the hidden layer. W_{ih} is the VxN matrix of weights between the input layer and the hidden layer of the network. W_{ho} is the NxV matrix of weights between the hidden layer and the output layer. The goal of the neural network training is to estimate these two matrices.

Given a word and assuming that $x_k = 1$ and $x_j = 0$ for $j \neq k$, we have:

$$h = X^T W_{ih} \qquad (2)$$

Thus, the hidden layer is obtained by a linear transformation of the input vector.

By deriving the error function on the output of the hidden layer, we obtain the updating equation for the *input to hidden weights*:

$$\frac{\partial E}{\partial \mathbf{h}} = \sum_{j=1}^{L(w)-1}\left(\sigma(\mathbf{v}_j'^T\mathbf{h}) - t_j\right)\cdot \mathbf{v}_j' \qquad (3)$$

where σ is a sigmoid function, t_j is *1* if the j-th node is the actual output word and v_j' is the j-th column of the weight matrix W_{ho}. $L(w)$ is the length of the path from the leaf w to the root in the tree for the hierarchical softmax.

For the case of updating the *hidden to output weights*, we obtain:

$$\frac{\partial E}{\partial \mathbf{v}_j'\mathbf{h}} = \sigma(\mathbf{v}_j'^T\mathbf{h}) - t_j \qquad (4)$$

Figure 2 shows a scheme of Mikolov's model on the standard case.

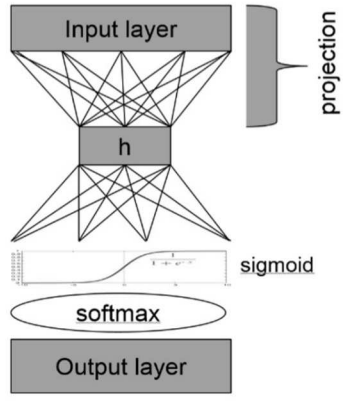

Figure 2: Mikolov's skip-gram neural network. Standard case.

Modified case

In the framework of neural networks, the sigmoid function allows to distort the representation space, so NNs can learn powerful non-linear transformations: in fact, with enough hidden units they can represent arbitrarily complex function. "By transforming the data non-linearly into a new space, a classification problem that was not linearly separable (not solvable by a linear classifier) can become separable." (Bengio *et al.*, 2015).

In this article, we propose to reinforce the non-linearity of the standard Mikolov's model. For this we add a sigmoid transformation between input layer and hidden layer (as in a classical MLP).

We chose the sigmoid function because it is a classical non-linear function used in neural network frameworks and its derivative is easy to compute:

$$\sigma(x) = \frac{1}{1+\exp(-x)} \qquad (5)$$

We add a non-linear transformation (sigmoid) to compute the hidden layer: the equation (2) is replaced by equation (6).

$$h = \sigma\,(X^T W_{ih}) \qquad (6)$$

We denote this case as *modified*: the hidden layer is obtained by a non-linear transformation of the input vector. As the model is more complex than the standard one, we hope that the new network is able to better model the relationships between words and so, the vector representation of words could be more accurate.

Figure 3 presents the proposed architecture. The sigmoid function is used to compute the hidden layer.

In the standard case (Fig. 2), at the end of the training, the hidden layer computed for a word will be used as projection (vector representation) for this word. In the modified case, we do not use the hidden layer as projection. Instead, we use the values before applying the sigmoid ($X^T W_{ih}$). So, the vector representation space keeps the linear structure as in the standard case (Mikolov's model). For this non-linear transformation, the derivatives of weights for the input to hidden layer are modified (equation (3)):

$$\frac{\partial E}{\partial \mathbf{h}} = \sum_{j=1}^{L(w)-1} \left(\sigma(\mathbf{v}'^{T}_j \mathbf{h}) - t_j\right) \cdot \mathbf{v}'_j \cdot h \cdot (1-h) \qquad (7)$$

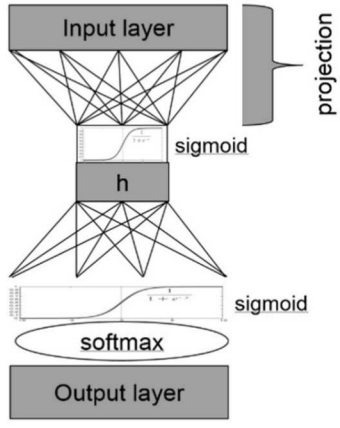

Figure 3. Mikolov's neural network. Modified case.

3. Experiments

In this article, ***selected PNs*** are the new proper names that we were able to find in DDs using our methods. ***Retrieved OOV PNs*** are the *selected PNs* that are present in the test documents. Using the DDs, we build a specific augmented lexicon for each test document according to the chosen period.

Results are presented in terms of *Recall* (%): the number of *retrieved OOV PNs* versus the number of *OOV PNs*. For the recognition experiments, *PN Error Rate* (PNER) is given. PNER is calculated like WER but taking into account only proper names. The best results are highlighted in bold in Tables.

3.1. Audio corpus

As audio corpus, seven audio documents of development part of ESTER2 (between 07/07/2007 and 07/23/2007) are used (Galliano *et al*, 2005). To artificially increase OOV rate, we have randomly removed 223 PNs occurring in the audio documents from our 122k ASR vocabulary. Table 1 summarizes the average occurrences of all PNs (IV and OOV) in audio documents with respect to 122k-word ASR vocabulary. Finally, the OOV PN rate is about 1.2%.

Corpus	Word occ	IV PNs	IV PN occ	OOV PNs	OOV PN occ
Audio	4525.9	99.1	164.0	30.7	57.3

Table 1. Average proper name coverage for audio corpus per file.

3.2. Diachronic corpus

French *GigaWord* corpus is used as diachronic corpus: newswire text data from *Agence France Presse* (AFP) and *Associated Press Worldstream* (APW) from 1994 to 2008. The choice of GigaWord and ESTER corpora was driven by the fact that one is contemporary with the other, their temporal granularity is the day and they have the same textual genre (journalistic) and domain (politics, sports, etc.).

3.3. Transcription system

ANTS (Illina *et al*., 2004) is based on Context Dependent HMM phone models trained on 200-hour broadcast news audio files. The recognition engine is Julius (Lee and Kawahara, 2009). The baseline phonetic lexicon contains 260k pronunciations for the 122k words. Using the SRILM toolkit (Stolcke, 2002), the language model is estimated on text corpora of about 1800 million words. The language model is re-estimated for each augmented vocabulary using the whole text corpora. The best way to incorporate the new PNs in the language model is beyond the scope of this paper.

4. Experimental results

4.1. Baseline results

We extract a list of all the OOV PNs occurring in the selected diachronic documents corresponding to the time period of the document to be transcribed. This period can be, for example, a day, a week or a month. After this, our vocabulary is augmented with the list of all extracted OOV PNs. If the diachronic corpus is large, a bad tradeoff between the lexical coverage and the increase of the lexicon size is obtained.

Using *TreeTagger* (Schmid, 1994), we extracted 160k PNs from 1 year of the diachronic corpus. Among these 160k PNs, 119k are not in our lexicon. Among these 119k, only 151 PNs are present in the audio corpus. It shows that it is necessary to filter this list of PNs to have a better tradeoff between the PN lexical coverage and the increase of the lexicon size.

Time period	Average of selected PNs per file	Average of retrieved OOV PNs per file	Recall (%)
1 day	532.9	10.0	32.6
1 week	2928.4	11.4	37.2
1 month	13131.0	17.6	57.2
1 year	118797.0	24.0	**78.1**

Table 2. Baseline results for audio corpus according to time periods. Values averaged on the 7 audio files.

Table 2 shows that using the DDs of 1 year, we extract, on average, 118797.0 PNs per file. Among these PNs, we retrieve on average 24.0 OOV PNs per audio file (compared to 30.7 in Table 1). This represents a recall of 78.1%.

4.2. NN-based results

We used Mikolov's open-source NN project available on the web. The NN is trained on the diachronic corpus described in Section 3.2 using only meaningful words. After preliminary experiments, we defined the best parameter set that will be used here: 400 for the size of the hidden layer, 20 for the context size and 5 for the number of negative samples. We performed 5 training epochs. Moreover, for the month time period, in order to

select more relevant PNs, a frequency threshold is used (the OOV PN occurring less than 3 times in the selected diachronic documents are excluded). We select the *Skip-gram* model because it achieved very good results on semantic tasks (Levy *et al.* 2015a). The target word is at the input layer and the context words are on the output layer.

An operating point of 15% of the average number of selected PNs per audio file seems to be a good compromise: 80 for day, 440 for week and 2000 for month (Fohr and Illina, 2015). This operating point is chosen to obtain a good recall with a reasonable number of *selected PNs*. This operating point will be used in our experiments.

In the first place, we analyze the effect of hierarchical softmax and negative sampling in the standard and modified cases. Table 3 shows the recall according to time periods. From the results it can be observed that for the standard case, hierarchical softmax gives better results for all periods (24.2% versus 22.3%, 32.1% versus 28.8% and 47.0% versus 44.2%). For the proposed modified case, negative sampling performs slightly better compared to hierarchical softmax. Comparing standard and modified cases, for a day and a month period, modified model is more powerful. For a week period this is not true. We are investigating why.

Method	1 day 80 selected PNs		1 week 440 selected PNs		1 month 2000 selected PNs	
	Std	**Modif.**	**Std**	**Modif.**	**Std**	**Modif.**
Hierarch softmax	24.2	25.1	**32.1**	31.6	47.0	47.4
Negative sampling	22.3	**25.6**	28.8	31.2	44.2	**48.8**

Table 3. Recall (%) for standard and modified cases according to time duration period for audio corpus. Values averaged on the 7 audio files.

At the end of the training, we obtain two weight matrices: word matrix W_{ih} and context matrix W_{ho}, according to the notations of Levy *et al.* (2015b). In Table 4, results are given for different time periods and different ways to calculate the projections using these two matrices. We want to analyze the importance of different projections. Negative sampling is only used because it obtains good results for modified case according to Table 3.

- ***ih only***: this method refers to Mikolov's word projection: for the word k it is the k-th row of the matrix W_{ih};
- ***ho only***: the projection of the word k is the k-th column of the matrix W_{ho};
- ***Concat***: the projection of the word k is the concatenation of the k-th row of the matrix W_{ih} and the k-th column of the matrix W_{ho};
- ***Sum***: in the standard and modified case, the representation space has a linear property: word vectors can be combined using vector addition. In this case, the projection of word k is the sum of the k-th row of the matrix W_{ih} and the k-th column of the matrix W_{ho}. Levy *et al.* (2015a) shown that using this kind of addition is equivalent to use first and second order similarities: "The second order similarity measures the extent to which two words are replaceable based to their tendencies to appear in similar contexts."

As shown in Table 4, these different projections only give tiny improvements over Mikolov's case or a degradation (*ho only*). *ho only* configuration uses only the context matrix and so the word representation is less accurate, as expected. *Sum* configuration seems to be neither good nor bad for our task and our corpus.

Method	1 day 80 selected PNs	1 week 440 selected PNs	1 month 2000 selected PNs
ih only	**25.6**	31.2	**48.8**
ho only	17.7	21.9	38.6
Concat (ih, ho)	25.1	31.6	**48.8**
Sum (ih+ho)	24.2	**32.6**	47.4

Table 4. Recall (%) for standard case according to time duration period for audio corpus. Negative sampling. Values averaged on the 7 audio files.

It confirms the results of Levy *et al.* (2015a): on eight datasets, results are improved in four cases and degraded for other four cases. So this behavior is perhaps corpus dependent or task dependent.

Time period	Method	Selected PNs	Recall (%)
1 day	Std HS	80	24.2
	Modif. NS	80	25.6
1 week	Std HS	440	32.1
	Modif. NS	440	31.2
1 month	Std HS	2000	47.0
	Modif. NS	2000	**48.8**

Table 5. Recall (%) for standard and modified cases according to time duration period for audio corpus. Hierarchical softmax (HS) for standard case and negative sampling (NS) for modified case. Values averaged on the 7 audio files.

Table 5, extracted from Table 3, summarizes the best results for standard and modified cases. It can be seen that the recall improvement is obtained with modified case for a day and a month periods. But, we notice that for a week period, the degradation is about 0.9%. A deeper analysis of this recall decrease is in progress.

4.3. Automatic speech recognition results for the audio corpus

We performed automatic transcription of the 7 audio documents using augmented lexicons (generating one lexicon per audio file, per period and per case). For generating the pronunciations of the added PNs, we used the G2P CRF approach (Illina *et al.*, 2011), trained on phonetic lexicon containing about 12000 PNs.

In order to incorporate the new PNs in the language model, we re-estimated it for each augmented vocabulary using the large text corpus described in Section 3.3. The number of selected PNs per period is the same as previously: 80 for a day, 440 for a week and 2000 for a month.

In terms of word error rate, no significant improvement is observed using the augmented lexicon. In this work, we are interested in the proper name recognition, so we compute also proper name error rate. Table 6 shows that, compared to standard lexicon, a significant improvement is obtained for the two NN systems (standard and modified cases) in terms of PN error rate (38.2, 38.3% versus 43.6%).

Stand. lexicon	Augmented lexicon			
	Method	**1 day 80 selected PNs**	**1 week 440 selected PNs**	**1 month 2000 selected PNs**
43.6	***Std*** ***HS***	40.9	40.4	38.3
	Modif. NS	40.5	40.4	**38.2**

Table 6. PNER (%) for standard and modified cases according to time duration period. Skip-gram model. Hierarchical softmax for standard case and negative sampling for modified case. Values averaged on the 7 audio files.

5. Conclusion

In this paper, the problem of OOV proper names and the vocabulary extension of a speech recognition system were investigated. Diachronic documents were used to retrieve new proper names and to enrich the vocabulary. We are interested on the continuous space word representation using neural networks proposed by Mikolov. One of the key contributions of this paper is to extend Mikolov's network by adding a non-linear transformation to better model the lexical and semantic context of proper names.

Experimental analysis on French corpus of broadcast news suggests the proposed modified configuration slightly outperforms the standard one in terms of recall. In terms of PNER, the results of both methods are comparable and show a significant decrease of the proper name error rate.

6. Acknowledgements

This work is funded by the *ContNomina* project supported by the French national Research Agency (ANR).

7. References

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

Bengio, Y., Goodfellow, I., Courville, A. (2015). Deep Learning. *Book in preparation for MIT Press.*

Church, K. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics,* 16(1):22–29.

Deng, L., Li, J., Huang, J.-T., Yao, K., Yu, D., Seide, F., Seltzer, M., Zweig, G., He, X., Williams, J., Gong, Y. and Acero A. (2013). Recent Advances in Deep Learning for Speech Research at Microsoft. *Proceedings of ICASSP*.

Federico, M. and Bertoldi, N. (2001). Broadcast News LM Adaptation using Contemporary Texts. *Proceedings of Interspeech*, pp. 239-242.

Fohr, D., Illina, I. (2015). Word Space Representations and their Combination for Proper Name Retrieval from Diachronic Documents. *Proceedings of Interspeech*.

Friburger, N. and Maurel, D. (2002). Textual Similarity Based on Proper Names. *Proceedings of the workshop Mathematical/Formal Methods in Information Retrieval*, pp. 155-167.

Galliano, S., Geoffrois, E., Mostefa, D., Choukri, K., Bonastre, J.-F., Gravier, G. (2005). The ESTER Phase II Evaluation Campaign for the Rich Transcription of French Broadcast News. *Proceedings of Interspeech*.

Illina, I., Fohr, D. and Linares, G. (2014). Proper Name Retrieval from Diachronic Documents for Automatic Transcription using Lexical and Temporal Context. *Proceedings of SLAM*.

Illina, I., Fohr, D., Jouvet, D. (2011). Grapheme-to-Phoneme Conversion using Conditional Random Fields. *Proceedings of Interspeech*.

Illina, I., Fohr, D., Mella, O., Cerisara, C. (2004). The Automatic News Transcription System: ANTS, some Real Time experiments. *Proceedings of. ICSLP*.

Kobayashi, A., Onoe, K., Imai, T., Ando, A. (1998). Time Dependent Language Model for Broadcast News Transcription and its Post-Correction. *Proceedings of ICSPL.*

Lee, A. and Kawahara, T. (2009) Recent Development of Open-Source Speech Recognition Engine Julius. *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC).*

Levy, O., Goldberg, Y., Dagan, I. (2015a). Improving Distributional Similarity with Lessons Learned from Word Embeddings. *In Transactions of the Association for Computational Linguistics, vol. 3.*

Levy, O. and Goldberg, Y. (2015b). Neural Word Embedding as Implicit Matrix Factorization. *Advances in Neural Information Processing Systems,* pp. 2177-2185.

Mikolov, T., Chen, K., Corrado, G. and Dean, J. (2013a). Efficient Estimation of Word Representations in Vector Space. *Proceedings of Workshop at ICLR*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. and Dean, J. (2013b). Distributed Representations of Words and Phrases and their Compositionality. *Proceedings of NIPS*.

Mikolov, T., Yih, W. and Zweig, G. (2013c). Linguistic Regularities in Continuous Space Word Representations. *Proceedings of NAACL HLT*.

Pennington, J., Socher, R., Manning, C. (2014). GloVe: Global vectors for Word Representation. *Proceedings of EMNLP.*

Schmid, H. (1994). Probabilistic Part-of-Speech Tagging using Decision Trees. *Proceedings of ICNMLP*.

Stolcke, A. (2002). SRILM - An Extensible Language Modeling Toolkit. *Proceedings of ICSLP*.

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