Uncertainty in Neural Network Word Embedding Exploration of Threshold for Similarity

Navid Rekabsaz, Mihai Lupu, Allan Hanbury[*]
Vienna University of Technology
Favoritenstrasse 9
Vienna, Austria
$family\_name$@ifs.tuwien.ac.at

ABSTRACT

Word embedding, specially with its recent developments, promises a quantification of the similarity between terms. However, it is not clear to which extent this similarity value can be genuinely meaningful and useful for subsequent tasks. We explore how the similarity score obtained from the models is really indicative of term relatedness. We first observe and quantify the uncertainty factor of the word embedding models regarding to the similarity value. Based on this factor, we introduce a general threshold on various dimensions which effectively filters the highly related terms. Our evaluation on four information retrieval collections supports the effectiveness of our approach as the results of the introduced threshold are significantly better than the baseline while being equal to or statistically indistinguishable from the optimal results.

1. INTRODUCTION

Understanding the meaning of a word (semantics) and of its similarity to the other words (relatedness) is the core of understanding text. An established method for quantifying this similarity is the use of *word embeddings*, where vectors are proxies of the meaning of words and distance functions are proxies of semantic and syntactic relatedness. Fundamentally, word embedding models exploit the contextual information of the target words to approximate their meaning, and hence their relations to the other words.

Given the vectors representing words and a corresponding mathematical function, word embedding models provide an approximation of the the relatedness of any two terms, although this relatedness could be perceived as completely-meaningless in the language. An emerging challenge here is: *how to identify whether the similarity score obtained from word embedding is really indicative of term relatedness?*. This issue is pointed out by Karlgren et al. [10] in examples, showing that word embedding methods are too ready to provide answers to meaningless questions: *"What is more similar to a computer: a sparrow or a star?"*, or *"Is a cell more similar to a phone than a bird is to a compiler?"*.

In the absence of a comprehensive answer, the need for related terms has been generally met by applying k Nearest Neighbours (k-NN) search such that retrieving the top k most similar terms in the neighbouring of a given term as related terms. Recently, Cuba Gyllensten and Sahlgren [3] point out the limitations of the k-NN approach as it neglects the internal structure of neighbourhoods which could be vastly different for various terms. In other words, some terms are more central in language and therefore have more related terms while many words have no genuinely related term. This is intuitive in human language while also quantifiable by using a language thesaurus e.g. WordNet (for example, by counting the number of synonyms). We therefore put the focus of this study on the notion of "similar" in word embedding.

Different characteristics of term similarities have been explored in several studies: the concept of relatedness [11, 13], the similarity measures [12], intrinsic/extrinsic evaluation of the models [1, 4, 21, 23], or in sense induction task [3, 5]. However, there is lack of understanding on the internal structure of word embedding, specifically how its similarity distribution reflects the relatedness of terms.

Following this direction, in this work, we would argue that the "similar" words can be identified by a threshold on similarity values which separates the semantically related words from the less or non-related ones. It is quite difficult, a priori, to even consider a threshold for this similarity. Especially since we do not want to make this parameter dependent on the term. This would be not only computationally, but also conceptually problematic. As Karlgren et al. discuss for the case of Random Indexing [9, 10], just because we *can* have a "most similar term(s)" does not mean that this makes any sense in real life.

Certainly, the meaning of "similar" also depends on the similarity function, but, we consider here the state of the art word similarity and leave the exploration of this factor for the further studies. Instead, we would argue that regardless of the similarity function, the most important factor is the threshold which separates the semantically related terms from the less or non-related ones.

Exploring such a threshold has the potential to bring improvements in those studies which use word embedding for retrieving the similar/related words in different tasks i.e. query expansion [7], query auto-completion [16], document retrieval [19], learning to rank [22], language modelling in IR [6], or Cross-Lingual IR [24].

We explore the estimation of this potential threshold by first quantifying the *uncertainty* factor in the similarity values of embedding models. This factor is an intrinsic characteristic of all the recent models, because they all start with some random initialization and eventually converge to a (local) solution. Therefore, even by training with the same parameters and on the same data, the created word embedding models result in slightly different word distri-

[*] This work is partly funded by two projects: SelfOptimizer (9867) by EuroStar and ADMIRE (P 25905-N23) by FWF. Thanks to Joni Sayeler and Linus Wretblad for their contributions in the SelfOptimizer project.

Neu-IR 16 SIGIR Workshop on Neural Information Retrieval July 21, 2016, Pisa, Italy

© 2016 Copyright held by the owner/author(s).

DOI:

butions and hence slightly different relatedness values. In the next step, using the *uncertainty* factor, we provide a continuous neighbouring representation for an arbitrary term, which is later used to estimate the general threshold.

In order to evaluate the effectiveness of the introduced threshold, we test it in the context of a document retrieval task, on five different test collections. In the experiments, we apply the threshold to identify the set of terms to meaningfully extend the query terms. We show that using the introduced threshold performs either exactly the same as or statistically indistinguishable from the optimal threshold for all collections.

In summary, the main contributions of the current study are:

1. exploration of the uncertainty factor in word embedding models in different dimensions and similarity ranges.
2. introducing a general threshold for separating similar terms in different dimensions.
3. extensive experiments on five test collections comparing different threshold values as well as k-NN search.

Among various word embedding models, in our study, we use the method proposed by Mikolov et al. [15]: skip-gram with negative-sampling training (SGNS) method in the Word2Vec framework. While this is not the newest method in this category (e.g. Pennington et al. [17] introduced GloVe and reported superior results), independent benchmarking provided by Levy et al. [14] shows that there is no fundamental performance difference between the recent word embedding models. In fact, based on their experiments, they conclude that the performance gain observed by one model or another is mainly due to the setting of the hyper-parameters of the models. Their study also motivates our decision to use SGNS: *"SGNS is a robust baseline. While it might not be the best method for every task, it does not significantly underperform in any scenario."*

The remainder of this work is structured as follows: First, we review related work in Section 2. We introduce the potential threshold in Section 3. We present our experimental setup in Section 4, followed by discussing the results in Section 5. Section 6 summarises our observations and concludes the paper.

2. RELATED WORK

The closest study to our work is Karlgren et al. [9], which explores the semantic topology of the vector space generated by Random Indexing. Based on their previous observations that the dimensionality of the semantic space appears different for different terms [10], Karlgren at al. now identify the different dimensionalities at different angles (i.e. distances) for a set of specific terms. It is however difficult to map these observations to specific criteria or guidelines for either future models or retrieval tasks.

In fact, our observations provide a quantification on Karlgren's claim that "'close' *is interesting and* 'distant' *is not"* [10].

More recently, Cuba Gyllensten and Sahlgren [3] follow a data mining approach to represent the terms relatedness by a tree structure. While they suggest traversing the tree as a potential approach, they evaluate it only on the word sense induction tasks and its utility for retrieving similar words remains unanswered. Our work complements and extends their approach. Defining the threshold on the collection and not each word, our method is efficiently applicable and computationally cheaper on all subsequent tasks to which the word embeddings may be applied.

3. POTENTIAL THRESHOLD

As mentioned in introduction, we are looking for a potential threshold to separate the truly related terms from the rest. In this section, we describe our analytic approach to explore such cutting points in different dimensions. The introduced threshold is defined on the entire model i.e. it is applicable to any arbitrary term in language.

For this purpose, we start with an observation on the uncertainty of similarity in word embedding models, followed by defining a continuous model of neighbouring distribution, before we define our proposed threshold.

3.1 Uncertainty of Similarity

In this section we make a series of practical observations on word embeddings and the similarities computed based on them.

To observe the uncertainty, let us consider two models P and M. To create each instance, we trained the Word2Vec SGNS model with the sub-sampling parameter set to 10^{-5}, context windows of 5 words, epochs of 25, and word count threshold 20 on the Wikipedia dump file for August 2015, after applying Porter stemmer. Each model has a vocabulary of approximately 580k terms. They are identical in all ways except their random starting point.

Figure 1a shows the distances between two terms and all other terms in the dictionary, for the two models, in this case of dimensionality 200. For each term we have approximately 580k points on the plot. As we can see, the difference between similarities calculated in the two models, appears (1) greater for low similarities, and (2) greater for a rare word (Dwarfish) than for a common word (Book). We can also observe that there are very few pairs of words with very high similarities.

Let us now explore the effect of dimensionality on similarity values and also uncertainty. Before then, in order to generalize the observations to an arbitrary term, we had to consider a set of "representative" terms. What exactly "representative" means is of course debatable. We took 100 terms recently introduced in the query inventory method by Schnabel et al. [21]. It is claimed that the terms are diverse in frequency, part of speech (POS). In the following of the paper, we refer to *arbitrary* term as an aggregation over the representative terms.

Figure 1b shows frequency histograms for the occurrence of similarity values in different dimensionalities of a given model. As we can see, similarities are in the $[-0.2, 1.0]$ range and have positive skewness (the right tail is longer). As the dimensionality increases, the kurtosis also increases (the histogram has thinner tails).

To observe the changes in uncertainty in different dimensions, we quantify this uncertainty as a function of the similarity value. Let us consider

$$\mathcal{S}_s = \{(x,y) : sim(\vec{x}_M - \vec{y}_M) \in (s, s+\epsilon)\}$$

the set of term pairs whose similarity is approximately s according to model M ($\vec{x}_M$ is the vector representation of term x in model M and sim is a similarity function between two vectors (Cosine throughout this paper)). We have to consider this approximation as it is practically never the case that two word pairs have exactly the same similarity value. We can then define an uncertainty ϱ as follows:

$$\varrho(s) = \frac{1}{|\mathcal{S}_s|} \sum_{(x,y)\in\mathcal{S}_s} |sim(\vec{x}_M, \vec{y}_M) - sim(\vec{x}_P, \vec{y}_P)| \quad (1)$$

where $\vec{x}_P$ is the vector representation of term x in model P. The approximation parameter ϵ is not important for this exemplification. For the plot in Figure 1c we take it to be 2.4×10^{-4}, as it splits our domain (-0.2,1.0) into 500 equal intervals. Figure 1c shows ϱ for different dimensionalities, against the similarity calculated in the M model. We observe that, as the similarity increases, the uncertainty decreases and that for highly similar words the different model instances tend to agree.

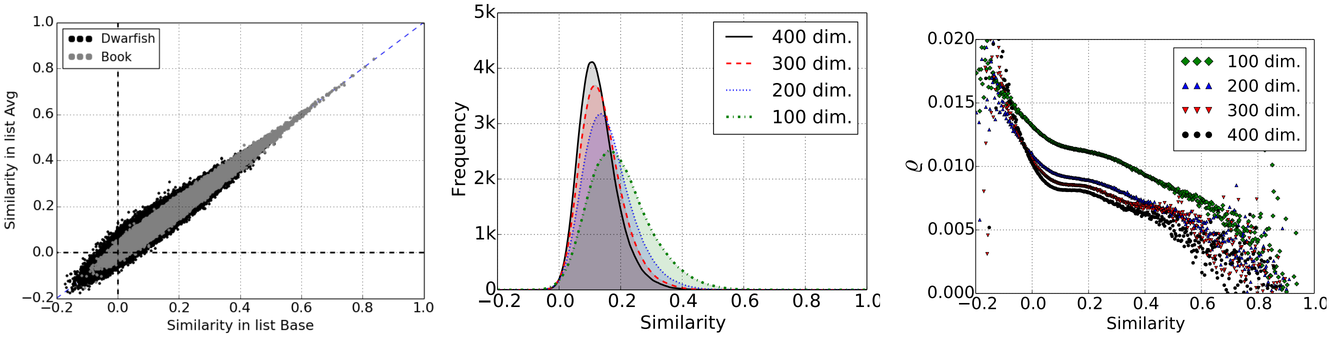


Figure 1: (a) Comparison of similarity values of the terms *Book* and *Dwarfish* to 580K words between models M and P. (b) Histogram of similarity values of an arbitrary term to all the other words in the collection for 100, 200, 300, and 400 dimensions. (c) Average uncertainty for each similarity value span.

We also observe a decrease in ϱ as the dimensionality of the model increases. On the other hand, the differences between models decrease as the dimension increases such that the models of dimension 300 and 400 seem very similar in comparison to 100 and 200. The observation shows a probable convergence in the Uncertainty at higher dimensionalities.

We can conclude from the observations that the similarity between terms is not a concrete value but can be considered as an approximation whose variation is highly dependent on the dimensionality and similarity range. We use the effect of this factor in the following.

3.2 Continuous Distribution of Neighbours

As seen, the different similarities of a pair of terms, achieved from different embedding models with the same training phases are slightly different. Intuitively, we assume that these similarity values follow a normal distribution such that we can consider every similarity value as a probability distribution, built based on the similarity values of the same pair in different models.

To estimate this probability distribution, for every dimension, we create five identical SGNS models following the setup in Section 3.1. Figure 2a shows the probability distribution of similarities for term *Book* to 25 terms in different similarity ranges[1]. We observe that by decreasing the similarity, the variation of the probability distributions increases, reflecting the increase in uncertainty empirically observed between two models in the previous section.

We use these probability distributions to provide a representation of the expected number of neighbours around an arbitrary term in the spectrum of similarity values. For this purpose, we calculate the mixture of cumulative distribution functions of the probability distributions subtracted from 1, showed in Figure 2b. The values on this plot indicate the number of expected neighbours in the area between the given similarity value to the term (similarity one). This representation of the expected number of neighbours in Figure 2b has two main benefits: (1) the estimation is continuous, and (2) it considers the effect of uncertainty and considers all the models.

As noted before, the notion of *arbitrary* term is in fact an average over the 100 representative terms. However, this are just a sample of all terms in the vocabulary. Therefore, in calculating the representation of the expected number of neighbours, we also consider the confidence interval around the mean. This interval is shown in Figure 2c. Here, the representation is zoomed on the lower left corner of Figure 2b. The area around each plot shows the confidence interval of the estimation.

Table 1: Potential thresholds

Dimensionality	Threshold Boundaries		
	Lower	**Main**	Upper
100	0.802	**0.818**	0.829
200	0.737	**0.756**	0.767
300	0.692	**0.708**	0.726
400	0.655	**0.675**	0.693

This continuous representation is used in the following for defining the threshold for the semantically highly related terms.

3.3 Similarity threshold

Given the representation of the expected number of neighbours around the arbitrary term, the question is "*what is the best threshold for filtering the highly related terms?*". This is of course a debatable question since the analytical approach attempts to measure the human understanding of synonymity. However, we hypothesise that since this general threshold tries to separates the highly related terms for an arbitrary term, it can be estimated from the average number of synonyms over the terms in language. Therefore, we transform the above question in a new question: "*What is the expected number of synonyms for a word in English?*"

To answer this, we exploit WordNet. We consider the distinct terms in the related synsets to a term as its synonyms, while putting out the multi word terms (e.g. Natural Language Processing, shown in WordNet by concatenating with underlines) since in creating the word embedding models we consider them as separated terms. The average number of synonyms over all the 147306 terms of WordNet is 1.6, while the standard deviation is 3.1.

Using the mean value, we define our threshold for each dimensionality as the point where the estimated number of neighbours in Figure 2c is equal to 1.6. We also consider an upper and lower bound for this threshold based on the points that the confident intervals cross the approximated mean. The results are shown in Table 1.

In the following sections, we validate the hypothesis by evaluating the performance of the introduced thresholds with an extensive set of IR experiments.

4. EXPERIMENTAL METHODOLOGY

We test the effectiveness of the potential threshold in an Ad hoc retrieval task on IR test collections by evaluating the results of applying various thresholds to retrieve the related terms.

Our relevance scoring approach is based on the *language model* [18] method as a widely used and established method in IR that has shown competitive results in various domains. In particular, we use the *translation language model* [2] which includes the similarity of

[1] we do not plot all to maintain the readability of the plot

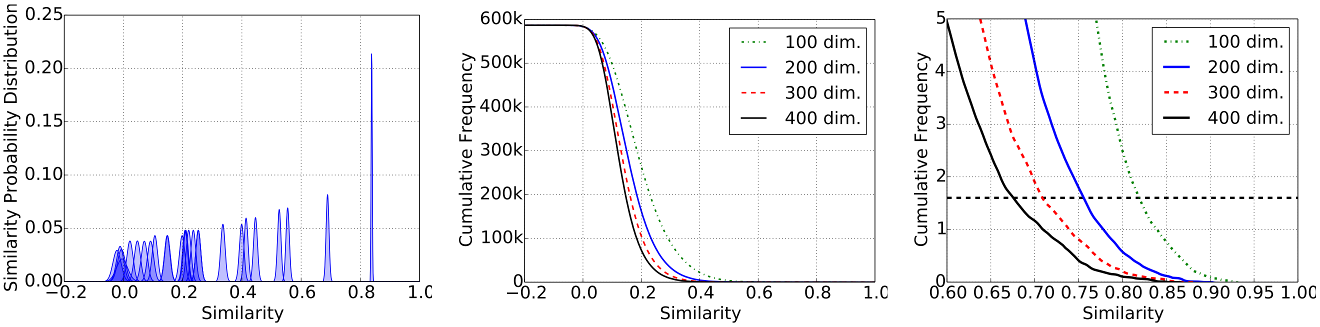


Figure 2: (a) Probability distribution of similarity values for the term *Book* to some other terms (b) Mixture of cumulative probability distributions of similarities in different dimensions (c) Expected number of neighbours around an arbitrary term with confidence interval. The average number of synonyms in WordNet (1.6) is shown by the dash-line.

related terms into the basic model.

In the following, first we explain the translation language model when combined with word embedding similarity and then describe the details of our experimental setup.

4.1 Translation Language Model

In the language model [18], the score of a document d with respect to a query q is considered to be the probability of generating the query by a model M_d estimated based on the document:

$$score(q,d) = P(q|M_d) = \prod_{t_q \in q} P(t_q|M_d) \tag{2}$$

Typically, the model is a multinomial distribution and the probability is computed with a maximum likelihood estimator, together with some form of smoothing. This smoothing, while not being part of the original idea, is in the practice of LM-based methods of paramount importance. However, this not being the focus of this study, we use Dirichlet smoothing [25], as many others have done, successfully, before us ([8,24,26]).

Berger and Lafferty [2] introduced translation models as an extension to the language modelling. A translation model introduces in the estimation of $P(q|M_d)$ a translation probability P_T, defined on the set of terms, always used in its conditional form $P_T(t|t')$ and interpreted as the probability of observing term t, having observed term t'.

$$P(q|M_d) = \prod_{t_q \in q} \left(\sum_{t_d \in d} P_T(t_q|t_d) P(t_d|M_d) \right) \tag{3}$$

The estimation of the model and specially the translation probability P_T have been addressed by various approaches during the last two decades. Recently, Zuccon et al. [26] integrates word embedding into the translation language model, showing potential improvement. In their work, they follow a k-NN approach to select the most similar terms for each query term in word embedding and estimate P_T based on the similarity of the extended terms to the query term.

Similar to their work, we use the translation language model enhanced with word embedding and reproduce some of their experiments. However, instead of the k-NN approach, we apply our introduced thresholds (Section 3.3) to filter the similar terms.

4.2 Experiments Setup

We evaluate our approach on 5 test collections: combination of TREC 1 to 3, TREC-6, TREC-7, and TREC-8 of the AdHoc track, and TREC-2005 HARD track. Table 2 summarises the statistics of the test collections. For pre-processing, we apply the Porter stemmer and remove stop words using a small list of 127 common English terms.

Table 2: Test collections

Name	Collection	# Doc
TREC 6	Disc4&5	551873
TREC 7, 8	Disc4&5 without CR	523951
HARD 2005	AQUAINT	1033461

In order to compare the performance of the potential thresholds, we test a variety of the threshold values in each dimension: for dimension 100, $\{0.67, 0.70, 0.74, 0.79, 0.81, 0.86, 0.91, 0.94, 0.96\}$, 200 dimension $\{0.63, 0.68, 0.71, 0.73, 0.74, 0.76, 0.78, 0.82\}$, 300 dimension, $\{0.55, 0.60, 0.65, 0.68, 0.70, 0.71, 0.73, 0.75\}$, and 400 dimension $\{0.41, 0.54, 0.61, 0.64, 0.66, 0.68, 0.70, 0.71, 0.75\}$. In addition to the threshold-based approach, we test the k-NN approach where N is tested with $\{1, 2, 3, 5, 7, 10\}$ values.

We set the basic language model as baseline and test the statistical significance of the improvement of all the results with respect to it (indicated by the symbol †). Since the parameter μ for Dirichlet smoothing of the translation language model is shared between the methods, the choice of parameters is not explored as part of this study. We select μ to 1000 as suggested in related studies. The statistical significance test are done using the two sided paired t-test and statistical significance is reported for $p < 0.05$.

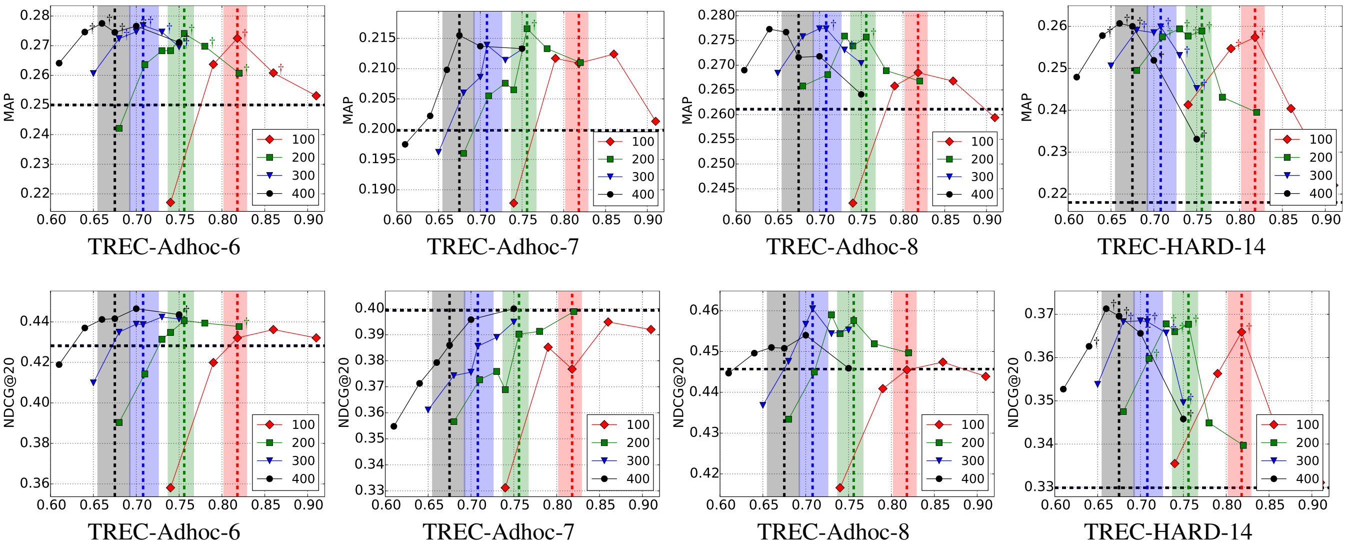
The evaluation of retrieval effectiveness is done with respect to MAP and NDCG@20, as standard measures. However, our initial experiments showed that using similar terms retrieved a substantial proportion of unjudged documents. Therefore, in order to provide a more fair evaluation framework, we consider MAP and NDCG over the condensed lists [20][2].

5. RESULTS AND DISCUSSION

The evaluation results of the MAP and NDCG@20 measures on the 4 test collections, with vectors in 100, 200, 300, and 400 dimensions are shown in Figure 3. For each dimension our threshold and its confidence interval are shown with vertical lines. Significant differences of the results to the baseline are marked on the plots using the † symbol. Table 3 summarizes the results of the optimal as well as potential thresholds.

Based on the results, we gain significantly better performance in all the collections at least in one of the threshold values. Except for TREC-7, we observe similar results with both the evaluations

[2] The condensed lists are used by adding the -J parameter to the trec_eval command parameters



TREC-Adhoc-6 TREC-Adhoc-7 TREC-Adhoc-8 TREC-HARD-14

TREC-Adhoc-6 TREC-Adhoc-7 TREC-Adhoc-8 TREC-HARD-14

Figure 3: MAP and NDCG@20 evaluation of the TREC-6, TREC-7, TREC-8 Adhoc, and TREC-2005 HARD for different thresholds and dimensions. Significance is shown by †. Vertical lines indicate our thresholds in different dimensions. To maintain visibility, points with very low performance are not plotted.

Table 3: In each cell, the top value shows the result of the potential threshold and the bottom reports the optimal value (shown as - when equal to our threshold value). † indicates a significant difference to the baseline. There is no significance difference between the results of the optimal value and our threshold.

Collection	100 (0.81)		200 (0.74)		300 (0.69)		400 (0.65)	
	MAP	NDCG	MAP	NDCG	MAP	NDCG	MAP	NDCG
TREC-6	0.273†	0.432	0.274†	0.441	0.277†	0.439	0.275†	0.442
	-	0.436	-	-	-	0.442	0.278†	0.447
TREC-7	0.211	0.377	0.217†	0.390	0.214	0.386	0.215	0.386
	0.212	0.395	-	0.399	-	0.395	-	0.400
TREC-8	0.269	0.446	0.276†	0.458	0.277†	0.461	0.272	0.451
	-	0.447	-	0.459	-	-	0.277	0.454
HARD	0.257†	0.366†	0.259†	0.368†	0.260†	0.368†	0.260†	0.370†
	-	-	-	-	-	-	0.261†	0.371†

measures.

The plots show that the performance of the method is highly dependent on the choice of the threshold value. In general, we can see a trend in all dimensions: the results tend to improve till reaching a peak (optimal threshold) and then decrease and finally converge to the baseline. Based on this general behaviour, we can assume that including the terms before the optimal threshold introduces noise and deteriorates the results while after it, the terms are filtered too strictly and there are still related terms to improve the results. Comparing the results of the optimal and potential threshold, in most the cases the optimal one is either the same or in the confidence area of our introduced threshold such that there is no statistically significant difference between the optimal and our threshold.

In order to have an overview on all the models, we calculate the gain of each model over the baseline and averaged the gains on the five collections. The results for MAP[3] are depicted in Figures 4a. Also the potential threshold and its confidence interval are compared with the optimal one in different dimensions in Figure 4b. Our threshold is optimal for dimensions 100, 200, and 300, and in dimension 400 it is statistically indistinguishable from the optimal. This results justifies the choice of the introduced threshold as a generally stable and effective cutting-point for identifying highly related terms.

For completeness, we also conducted experiments on the k-NN approach. The results in Figure 4c show the very weak performance of the k-NN approach for MAP measure such that it has slightly better than baseline for k equal to 1 and 2 and then radically deteriorates by increasing k.

Table 4: Examples of similar terms, selected with the potential threshold

book: publish, republish, foreword, reprint, essay
eagerness: hoping, anxious, eagerness, willing,wanting
novel: fiction, novelist, novellas, trilogy
microbiologist: biochemist, bacteriologist, virologist
shame: ashamed
guilt: remorse
Einstein: relativity
estimate, dwarfish, antagonize: no neighbours

To understand this behaviour let us take a closer look at the selected terms. Table 4 shows some examples of the retrieved terms when using the word embedding model with 300 dimension with the our threshold (same as optimal in this dimension). The examples show the strong differences in the number of similar words for various terms. The mean and standard deviation of the number of similar terms for the 508 query terms of the tasks is 1.5 and 3.0 respectively. Almost half of the terms are not expanded at all. An interesting observation is the similarity between this calculated mean and standard deviation and the aggregated number of synonyms we observed in WordNet in Section 3.3—mean of 1.6 and standard deviation of 3.1. It appears that although the two semantic resources cast the notion of similarity in very different ways and their provided sets of similar terms are very different, they correspond to

[3] The NDCG results are very similar and not shown for space

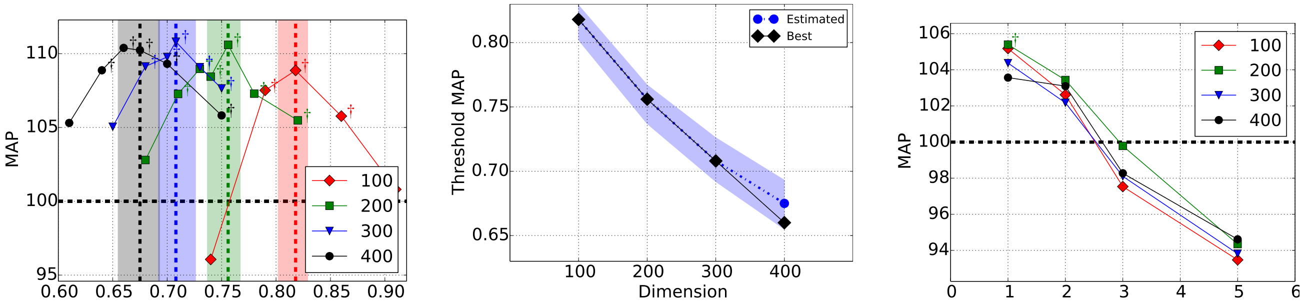


Figure 4: All the results in MAP measure: (a,c) Improvement of the models with respect to the original language model (baseline), aggregated over all the collections (b) The potential and optimal thresholds in different dimensions where results are aggregated over all the collections. (c) same as Figure a but using k-NN approach.

very similar distribution of the number of related terms.

6. CONCLUSION AND FUTURE WORK

We have analytically explored the thresholds on similarity values of word embedding to select related terms. This threshold is estimated based on a novel representation of the neighbours around an arbitrary term which is continuous and benefits from addressing the issue of uncertainty in similarity values of modern word embedding models.

We extensively evaluate the application of the suggested threshold on four information retrieval collections. The results show superior performance when using our threshold such that its results are either equal to or statistically indistinguishable from the optimal results, achieved by extensive search on the parameter space.

7. REFERENCES

[1] M. Baroni, G. Dinu, and G. Kruszewski. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proc. of ACL Conference*, 2014.

[2] A. Berger and J. Lafferty. Information Retrieval As Statistical Translation. In *Proc. of SIGIR*, 1999.

[3] A. Cuba Gyllensten and M. Sahlgren. Navigating the semantic horizon using relative neighborhood graphs, 2015.

[4] L. De Vine, G. Zuccon, B. Koopman, L. Sitbon, and P. Bruza. Medical semantic similarity with a neural language model. In *Proc. of CIKM*.

[5] K. Erk and S. Padó. Exemplar-based models for word meaning in context. In *Proc. of ACL*, 2010.

[6] D. Ganguly, D. Roy, M. Mitra, and G. J. Jones. Word Embedding based Generalized Language Model for Information Retrieval. In *Proc. of SIGIR Conference*, 2015.

[7] M. Grbovic, N. Djuric, V. Radosavljevic, F. Silvestri, and N. Bhamidipati. Context-and content-aware embeddings for query rewriting in sponsored search. In *Proc. of SIGIR Conference*, 2015.

[8] M. Karimzadehgan and C. Zhai. Estimation of Statistical Translation Models Based on Mutual Information for Ad Hoc Information Retrieval. In *Proc. of SIGIR*, 2010.

[9] J. Karlgren, M. Bohman, A. Ekgren, G. Isheden, E. Kullmann, and D. Nilsson. Semantic topology. In *Proc. of CIKM Conference*, 2014.

[10] J. Karlgren, A. Holst, and M. Sahlgren. Filaments of meaning in word space. In *Proc. of ECIR Conference*, 2008.

[11] D. Kiela, F. Hill, and S. Clark. Specializing word embeddings for similarity or relatedness. In *Proc. of EMNLP*, 2015.

[12] B. Koopman, G. Zuccon, P. Bruza, L. Sitbon, and M. Lawley. An evaluation of corpus-driven measures of medical concept similarity for information retrieval. In *Proc. of CIKM*, 2012.

[13] G. Kruszewski and M. Baroni. So similar and yet incompatible: Toward automated identification of semantically compatible words. In *Proc. of NAACL*, 2015.

[14] O. Levy, Y. Goldberg, and I. Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transaction of the Association of Computational Linguists*, 2015.

[15] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[16] B. Mitra. Exploring session context using distributed representations of queries and reformulations. In *Proc. of SIGIR Conference*, 2015.

[17] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. *Proc. of EMNLP Conference*, 2014.

[18] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proc. of SIGIR*, 1998.

[19] N. Rekabsaz, R. Bierig, B. Ionescu, A. Hanbury, and M. Lupu. On the use of statistical semantics for metadata-based social image retrieval. In *Proc. of CBMI Conference*, 2015.

[20] T. Sakai. Alternatives to bpref. In *Proc. of SIGIR*, 2007.

[21] T. Schnabel, I. Labutov, D. Mimno, and T. Joachims. Evaluation methods for unsupervised word embeddings. In *Proc. of EMNLP*, 2015.

[22] A. Severyn and A. Moschitti. Learning to rank short text pairs with convolutional deep neural networks. In *Proc. of SIGIR*, 2015.

[23] Y. Tsvetkov, M. Faruqui, W. Ling, G. Lample, and C. Dyer. Evaluation of word vector representations by subspace alignment. In *Proc. of EMNLP*, 2015.

[24] I. Vulić and M.-F. Moens. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proc. of SIGIR*, 2015.

[25] C. Zhai and J. Lafferty. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proc. of SIGIR*, 2001.

[26] G. Zuccon, B. Koopman, P. Bruza, and L. Azzopardi. Integrating and evaluating neural word embeddings in information retrieval. In *Proc. of Australasian Document Computing Symposium*, 2015.