On the effect of word frequency on distributional similarity

Christian Wartena
Hochschule Hannover
Department of Information and Communication
Expo Plaza 12, 30539 Hannover, Germany
Christian.Wartena@hs-hannover.de

Abstract

The dependency of word similarity in vector space models on the frequency of words has been noted in a few studies, but has received very little attention. We study the influence of word frequency in a set of 10 000 randomly selected word pairs for a number of different combinations of feature weighting schemes and similarity measures. We find that the similarity of word pairs for all methods, except for the one using singular value decomposition to reduce the dimensionality of the feature space, is determined to a large extent by the frequency of the words. In a binary classification task of pairs of synonyms and unrelated words we find that for all similarity measures the results can be improved when we correct for the frequency bias.

1 Introduction

Distributional similarity has become a widely accepted method to estimate the semantic similarity of words by analyzing large amounts of texts. The basic idea of distributional similarity is that words occurring in similar contexts have a similar meaning. However, implementations of the idea differ by choosing different features to represent the context of a word, by different approaches to determine feature weights and by different similarity measures to compare the contexts. A number of recent studies (Bullinaria and Levy, 2007; Bullinaria and Levy, 2012; Kiela and Clark, 2014) shed light on the influence of a number of design choices on the performance of distributional similarity in various tasks.

Usually it is assumed that a minimum number of occurrences of a word is needed to build a reliable distributional model of the word. Ferret (2010) e.g. observes that results become significantly worse when less than 100 occurrences of a word are available.

Besides the fact, that a minimum number of occurrences is required to get any reliable information about a word at all, another problem is the fact that similarity measures tend to have a frequency bias. Weeds et al. (2004) evaluated a number of combinations of feature weighting schemes and similarity measures and found that each combination has a frequency bias: when we look for the words that are most similar to a given word, most measures prefer more frequent words. A few measures have a bias towards less frequent words or words with a frequency similar to the target word. The larger the difference in frequency between the most frequent and the least frequent word included in some test set is, the stronger the influence of the frequency bias will become. Thus the frequency bias poses a further burden upon the inclusion of infrequent words in a task.

Experiments in which the quality of distributional methods is tested usually involve many words for which information in lexical resources is available and that occur quite frequently in large corpora. However, if we look at the distribution of words in a corpus the vast major-

ity of words occurs only very rarely. E.g. according to Barroni et al. (2009) the large ukWaC corpus contains about $1.529 \cdot 10^6$ different word forms tagged as common noun by the TreeTagger (Schmid, 1995), $1.414 \cdot 10^6$ of which occur less than 20 times. In most studies a minimum number of 20, 100 or sometimes even 1000 occurrences of a word is assumed to be necessary to compute reliable similarities. Thus for most words distributional similarity cannot be used.

One of the practical applications of distributional similarity that is often mentioned, is automatic updating and extension of a thesaurus with new terminology (Crouch, 1990; Curran and Moens, 2002; Turney and Pantel, 2010). One of the typical properties of new terminology is, that we do not yet have many occurrences of the terms in our corpus. Thus, the methods developed are in fact not suited for this useful application. For many other applications a similar situation holds. Thus, if we want to make distributional similarity more useful for applications, we need to improve the way we can deal with infrequent words. Before we can improve methods for infrequent words, we need to better understand, how various implementations of distributional similarity depend on word frequency.

In the present paper we study the frequency bias in more detail for 6 different similarity methods. First we compare the methods on a standard task, the synonymy task that has been included in the Test of English as a Foreign Language (TOEFL). In two experiments we then compute the similarity of pairs of English words with different frequencies using the ukWaC corpus. In the first experiment we compute the similarity of 10 000 arbitrary word pairs in which the frequency of the first word is kept constant and the frequency of the second word varies. In this experiment we can observe for each method, how the similarity depends on the word frequency. In the second experiment we investigate the behavior of the methods in a task in which 10 000 pairs of synonyms and non-synonyms have to be ranked. For this test a set of word pairs was used that was selected from Wordnet without putting restrictions on the frequency of the involved words in some corpus. Finally, we show how much the results of each method can be improved by taking into account the similarity expected on the base of the frequency of the words.

In section 2 we discuss related work. In section 3 we present the details of the distributional methods compared. Section 4 describes the data and the experiments used to study the influence of word frequency on word similarity for each method. The results of the experiments are given and discussed in section 5.

2 Related Work

Despite the importance of being able to deal with infrequent words, the problem has received very little attention. Ferret (2010) computes the similarity of huge amounts of word pairs in order to extract synonyms from a mid-sized corpus. He systematically investigates the results for low frequent, mid frequent and highly frequent words using cosine similarity and pointwise mutual information for feature weighting. He concludes that the results for the low frequent words (less than 100 occurrences) are useless.

Kazama et al. (2010) propose a method to extract word pairs with a high likelihood to be semantically related. They argue that, given two word pairs with the same (distributional) similarity, the pair with more frequent words should become a higher likelihood to be semantically related. The rationale behind this is, that we have more observations and thus a more reliable estimation of the similarity. Thus their method becomes robust when dealing with sparse data. However, if the task is not to extract pairs of related words from a corpus, but to decide whether two given words are related or not, we do not want to decide that the words are unrelated just by the fact that we do not have enough observations.

Already Patel et al. (1998) found a clear correlation between the frequency of words and their similarity. However, they were more interested in corpus size than in word frequencies. As mentioned above, Weeds et al. (2004) study the frequency bias for several methods in the case that similar words for a given word are sought. They do not consider the direct dependency of the similarity values on the frequency of the words, but study the frequency of the most similar words that are found, in relation to the frequency of the target word.

In two previous studies we investigated the dependency of the similarity of a pair of two words on the frequency of these words (Wartena, 2013a; Wartena, 2013b). In these studies we could improve the results for two different tasks substantially by using the difference between the similarity predicted by the frequency of the words and the actual measured similarity. However, in both studies only random indexing was considered. Random indexing is an efficient method for dimensionality reduction using random projection of features into a small size feature space. However, the results using random indexing are probably not as good as those obtained with other dimensionality reduction methods and the method is not very popular in the field of distributional semantics. In the present study we extend the previous studies and also include other similarity measures and different feature weighting schemes.

3 Overview of used similarity methods

The computation of distributional similarity of two words always involves two steps: first distributional models for each word are built by collecting information about the contexts in which the words occur. Subsequently these models are compared to access the similarity of the words. As the models are usually vectors in a high dimensional feature space or probability density distributions a number of well known similarity measures can be used. Also for the construction of the models a number of choices has to be made: it has to be decided which context information is used; several possibilities exist for the weighting of the context features and finally some dimensionality reduction techniques might be applied.

In order to study the effects of word frequency on distributional similarity it is not feasible to explore all possible combinations of choices for context features, weighting method, dimensionality reduction technique and similarity measure. Fortunately, a few recent studies have investigated the effect of various design choices and combinations of choices for different tasks and corpora in a systematic way (Bullinaria and Levy, 2007; Bullinaria and Levy, 2012; Kiela and Clark, 2014). In the present study we will include a number of methods that turned out to be successful in the mentioned studies.

In the simplest case we use just the frequencies of context words as a feature vector in combination with cosine similarity. We will refer to this configuration as *plain_cos*. We also include the same method in combination with the Jensen-Shannon divergence, which we call *plain_jsd*. A successful weighting scheme turned out to be pointwise mutual information (PMI) between a word and a feature. As it makes no sense to use an information theoretical measure like Jensen-Shannon divergence (JSD) for weighted features, we use PMI only in combination with cosine similarity and refer to this combination as *plain_pmi*. We also consider the variant where the feature space of the last method is reduced using singular value decomposition (*svd*). Alternatively, we use random indexing to reduce the feature space. We use random indexing both in combination with cosine similarity (*ri_cos*) and with JSD (*ri_jsd*)

In the following we will discuss the various parameters for each configuration in more detail.

3.1 Context features

As context features we use the lemmata of words in the context window of the target word. Sometimes a combination of a word and its syntactic relation to the target word is used. However, it is not clear whether inclusion of syntactic dependencies systematically improves the quality of the feature vectors (Giesbrecht, 2010; Kiela and Clark, 2014). Both Bullinaria and Levy (2012) and Kiela and Clark (2014) show that lemmatization always improves the results, though both studies do not agree about the effect of stemming. Here we use in all cases the lemmata as context features, but we compute the context models for surface forms of the words. Thus, we never lemmatize the words in the test sets. For the method that uses singular value decomposition (SVD) we have to include context vectors of words that are not part of the test. For these additional words we use lemmata as well.

Inclusion of function words and other highly frequent words put a heavy load on all subsequent computations and might even have a negative effect on the performance (Bullinaria and Levy, 2012). Thus we decided to exclude all closed-class words (determiners, conjunctions, prepositions, etc.). Furthermore we exclude all words

from a small standard stop word list (taken from Lucene). After removal of these words we take two words to the left and to the right of each word within the same sentence as context features.

Very infrequent words do not contribute very much to the context vectors. Thus, after selecting the context words, we remove those words that fall outside a given frequency range in the corpus. For all experiments we use the ukWaC corpus (Baroni et al., 2009). For the conditions *plain_cos*, *plain_jsd*, *plain_pmi* and *svd* we kept only words that occur at least 5000 times and at most 1 000 000 times in this corpus in the first experiment. This gives us 16 617 context words that are used as features. In the random word pair experiment and in the synonym ranking task (both involving much more words for which context vectors have to be computed) we kept words occurring at least 10 000 times and at most 1 000 000 times, resulting in a set of $10\,800$ context features. For random indexing using much more words is no problem and also improved the results in preliminary experiments. Thus we take words in the frequency range from 5 to 1 000 000 occurrences, resulting in 935 405 different features.

3.2 Feature weighting

Positive Pointwise Mutual Information (PPMI) is a popular feature weighting scheme and it was shown both in the studies of Bullinaria and Levy (2012) and of Kiela and Clark (2014) that PPMI in combination with several similarity measures gives optimal results. PPMI is defined as the maximum of 0 and the pairwise mutual information. We use the PPMI for feature weighting in *plan_pmi* and *svd*. For all other configurations raw feature counts are used.

3.3 Dimensionality Reduction

Given the huge amount of different words that can appear in the context of a word, we always will end up with very high dimensional and very sparse feature spaces. Therefor, often some form of dimensionality reduction technique is used. Moreover, techniques like singular value decomposition (SVD) will find the most important underlying factors determining the use of a word and separate them from less important factors that are probably not related to the meaning of the word.

We use SVD in one condition. First we construct the full co-occurrence matrix of $16\,617 \times 16\,617$ with almost $78 \cdot 10^6$ non-zero entries for the TOEFL-Test. In the random word pair experiment and in the synonym ranking task, in which we used less context features, the size of the matrix is $20\,788 \times 10\,800$ and $18\,145 \times 10\,800$, respectively. Subsequently we compute the positive pairwise mutual information (PPMI) for each word/feature pair and adjust the values in the co-occurrence matrix. Using the svdlib library from the semantic vectors package (`https://code.google.com/p/semanticvectors/`) we compute matrices U, S and V such that $M = USV^T$, where U and V are orthogonal matrices and S is a diagonal matrices of the singular values of M where M is the original word-lemma matrix of PPMI values. We now can use the rows of US as feature vectors for the words. By truncating the rows we can restrict the comparison of the feature vectors to the most important principle components. We will use the first 5 000 components in the experiments below.

Bullinari and Levy (2012) found that results can be improved when the influence of the first components is reduced. To do so, they either simply leave out the first n principal components or reduce the weights of the most important features by using the matrix $X = US^P$ instead of $X = US$, where P is called Caron's P. Following Bullinaria and Levy we use a value of $0,25$ for Caron's P.

An alternative way to reduce the number of dimensions is random projection. Random projection was introduced for distributional similarity by Karlgren and Sahlren (2001) under the name random indexing. Random indexing has the great advantage that it is computationally very cheap and there is no need to build the full co-occurrence matrix. Each feature is represented by a n-dimensional vector with a 1 at k random positions and 0 at all other positions. In the following we set $n = 10\,000$ and $k = 10$. This vector can be seen as a *fingerprint* of the feature. The context of a word is represented by the sum of the vectors of all words found in its context.

The advantage of this method is that the number of dimensions can be chosen freely and no additional computation for dimension reduction is

needed. Random Indexing is not used very widely and not included in a number of overview studies. However, Random Indexing was shown to yield competitive results at the 2013 Semeval phrasal semantics task (Korkontzelos et al., 2013).

3.4 Similarity Measures

Various similarity measures have been used for distributional semantics. If we use vectors of simple word occurrences cosine similarity is an obvious choice. In the studies of Bullinaria and Levy (2007) and of Kiela and Clark (2014) this measure performed very well in combination with various weighting schemes and for various tasks. We use cosine similarity for the conditions *plain_cos*, *plain_pmi*, *svd* and *ri_cos*.

Alternatively, we can see the distributional model of a word as a probability distribution over words that can appear in the context of that word. Then it is natural to use a information theoretic similarity measure. Since we usually want a symmetric measure the most commonly used measure is the Jensen Shannon Divergence (JSD). JSD was shown to give also very good results, especially in combination with unweighted features (Bullinaria and Levy, 2007; Kiela and Clark, 2014). We use JSD in the conditions *plain_jsd* and *ri_jsd*.

4 Data and Experiments

For all experiments described below we compute the context vectors on the ukWaC-Corpus (Baroni et al., 2009). First we examine how each method performs on the widely used TOEFL synonym test. Then we study the influence of word frequency on a set of 10 000 randomly selected word pairs. Finally, we compare the methods is a test in which 10 000 pairs of synonyms and unrelated words have to be ranked.

4.1 TOEFL Synonym Test

One of the most widely used tests to evaluate semantic similarity is the synonymy task that has been included in the Test of English as a Foreign language (TOEFL) (Landauer and Dumais, 1997). The test consists of 80 words and for each word four potential synonyms. In total 391 words are involved. The task is to decide which of the four candidates is the synonym. When we choose always the candidate with the largest distributional similarity, we see how well the chosen measure reflects semantic similarity. We include this test to get an impression of the quality of the methods included in the following experiments.

4.2 Random Word Pairs Experiment

For our first experiment to access the behavior of the similarity measures for words with different numbers of observations we have extracted 10 000 word pairs from the ukWaC corpus in the following way: we selected 100 words that occur at least 1000 and at most 1005 times in the corpus and that have a part-of-speech tag from an open word class, consist of at least 3 letters and do not contain special characters. These words are used as the first component of the word pairs. Next we randomly selected 10 000 words from the corpus with the same criteria but in a frequency range from 5 to 1 000 000. This was done by ordering all words according to their frequency and picking words with a fixed interval from that list. Thus the frequency distribution of these words is the same the that of all words in the corpus. Finally, these 10 000 words were assigned to the previously selected words to obtain 10 000 word pairs.

For these pairs we compute the similarity for each method. In order to see to what degree the similarity of a pair of words depends on the frequency of the words, we predict the similarity for each pair by taking the average similarity of 100 word pairs with the same or almost the same frequency. To do so, we order the all pairs according to the frequency of their second word (the frequency of the first word of each pair is always the same) [1]. Now we compute the average similarity of 50 pairs before and 50 pairs after the pair under consideration. Finally, we compute the coefficient of determination as follows:

$$R^2 = 1 - \frac{\sum_i (sim_i - \overline{sim_i})^2}{\sum_i (sim_i - \overline{sim})^2} \qquad (1)$$

where sim_i is the found similarity of the i-th pair, $\overline{sim_i}$ predicted similarity (moving average) for that pair and $\overline{sim}$ is the average similarity of all pairs.

[1] In case the the frequencies of the second word are identical, we order the pairs alphabetically. However, any other ordering did not influence the results presented below within the precision of two decimals.

4.3 Synonym Ranking Task

In the last experiment we want to investigate how much each method can be improved when we correct for the frequency bias.

Association tasks in which a word has to be associated with one word from a small list of words, have been used in many studies on distributional similarity. However, for some applications we are confronted with a completely different situation. A possible application is to add terminology extracted from a corpus to an existing thesaurus. Each term now is either a synonym of one of many thesaurus terms, or it is new concept for which no synonyms are present in the thesaurus. In fact for each pair we have to decide whether the words are synonym or not.

Another problem of the TOEFL test and some other tests is the small size: the TOEFL set has 80 pairs, the Rubinstein-Goodenough set consists of 65 pairs (Rubenstein and Goodenough, 1965), the Finkelstein's WordSim-353 set consists of 353 pairs (Finkelstein et al., 2001). Moreover, some data focus more on word associations than on synonymy. Finally, many larger generated data sets have a strong frequency bias. E.g. for their Wordnet Based Similarity Test, with questions similar to those from the TOEFL test, Freitag et al. (2005) have chosen only words occurring at least 1000 times in the North American News corpus (about 1 billion words); for a lexical entailment task Zhitomirsky- Geffet and Dagan (2009) use only words occurring at least 500 times in a 18 Million word corpus; for their distance comparison Bullinaria and Levy (2007) select 200 words "that are well distributed in the corpus" and the test set for two word phrases constructed by Mitchell and Lapata (2010) consists of phrases occurring at least 100 times in the British National Corpus (100 million words).

In an application in which e.g. new terminology has to be mapped onto an existing thesaurus, we do not want to exclude infrequent words. In contrary: the new and rare words are the most interesting ones. Therefor we use in our last experiment a data set of almost 10 000 word pairs in which no infrequent words are excluded. We have used this data set before in a similar experiment (Wartena, 2013a)[2]. This list of pairs consists of single words taken from Wordnet (Miller, 1995) that occur at least two times in the British National Corpus and at least once in the ukWaC corpus. The data set contains 849 pairs for which the Jaccard coefficient of the sets of Wordnet senses of the words is at least 0.7. These word pairs are considered to be synonyms. As non-synonyms 8967 word pairs are included that share no senses.

The task now is to decide for each pair, whether the words are synonym or not. We evaluate similarity measures for this task by ranking the pairs according to the similarity of the words. An ideal ranking, of course, would put all synonyms on top. To what extent this is the case is indicated by the area under the ROC curve (AUC).

For the pairs in this data set we also want to predict the similarity using the word frequency. The situation is a bit more complicated than before, since the frequency of both words is variable. Here we follow our previous finding that the similarity is determined mostly by the frequency of the least frequent word (Wartena, 2013b). We thus take the moving average of the similarity when the pairs are ordered according to the minimum of the word counts as prediction. Finally, we rank the pairs according to their residual values, assuming that a pair is likely to be semantically related if the observed distributional similarity is larger than we would expect from the frequency of the words.

5 Results

Though our implementation of random indexing is not exactly the same as that described by Karlgren and Sahlren (2001) (e.g. we so not use lower weights for more distant words) and though we use a different corpus, we get the same result on the TOEFL synonym task. Best results are obtained using SVD. However, the results fall clearly back behind those obtained by Bullinaria and Levy (2012), despite the fact that we roughly made the same choices for all parameters.

The results of the random word pair experiment are given in Table 2. The similarities based on SVD are almost independent of the frequency of

[2]The data set is available at `http://nbn-resolving.de/urn:nbn:de:bsz:960-opus-4077`.

Table 1: Results of 6 different distributional similarity methods on the TOEFL synonym task using the ukWaC- Corpus

Method	Fraction correct
plain_cos	0.675
plain_jsd	0.688
plain_pmi	0.788
svd	0.863
ri_cos	0.725
ri_jsd	0.650

Table 2: Dependency of 6 different distributional similarity methods for 10 000 random pairs of words on the frequency of the words.

Method	R^2
plain_cos	0.20
plain_jsd	0.77
plain_pmi	0.47
svd	0.10
ri_cos	0.39
ri_jsd	0.87

the words. Especially the similarities computed using the Jensen-Shannon divergence are highly determined by the frequency. Interestingly we see that the R^2 value for *plain_pmi* is much larger than for *plain_cos*. The dependency of the similarity on the frequency of the second word is illustrated exemplary in Figure 1 and 2 for the configurations *plain_pmi* and *ri_cos*. We see that the moving average for the methods using cosine similarity is roughly logarithmic function of the frequency. For the JSD the moving average follows a kind of asymmetric sigmoid curve.

In the synonym ranking task (Table 3) we do not find any surprises: as in the case of the TOEFL-test the best result is obtained with the *svd*-configuration, the second best with *plain_pmi* and results based on the Jensen-Shannon divergence are worst. The dependency on the word frequency, measured by coefficient of determination, also confirms the results of the previous experiment, though the absolute values are a bit different. Remarkable, however, are the results of the ranking by the residual values. The results of all methods could be improved. The largest improvements, of course, are found for the methods with the largest dependency on the word frequency. The differences between the methods now become much smaller. The methods *svd* and *plain_pmi* now give the same results.

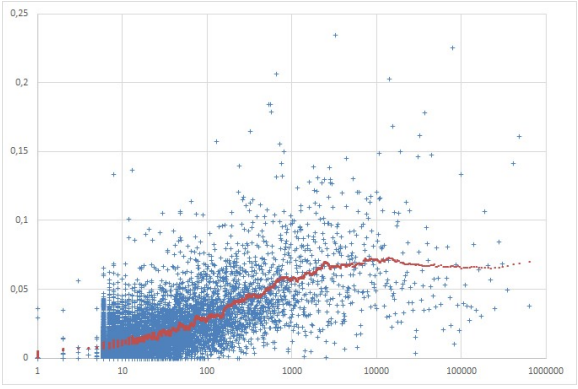

Figure 1: Similarity of wordpairs using the *plain_pmi* configuration in dependence of the frequency of the second word. The first word in each pair always occurs between 1000 and 1005 times in the corpus. The y-axis is represents the cosine similarity, the x-axis the number of occurrences of the second word. The solid (red) line is the moving average in a window of 100 word pairs.

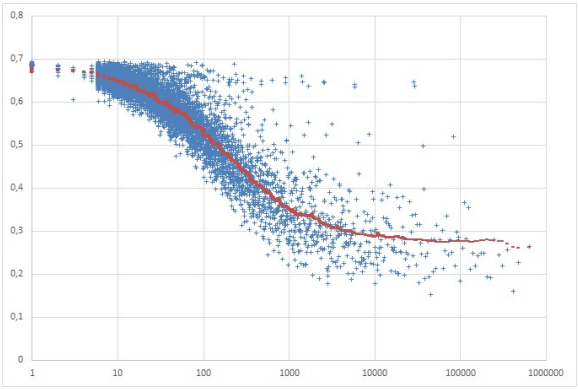

Figure 2: Similarity of wordpairs using the *ri_jsd* configuration in dependence of the frequency of the second word. The y-axis is represents the Jensen-Shanon divergence of the context vectors of the words, the x-axis the number of occurrences of the second word. The solid (red) line is the moving average in a window of 100 word pairs

Finally, we also want to know how the 6 methods perform for word pairs involving an infrequent word. Table 4 gives the results for all pairs with at least one word occurring less than 100 times in the ukWaC corpus. We observe that the results for the methods that have a strong fre-

Table 3: Results of 6 different distributional similarity methods on ranking 10 000 pairs of synonyms and non-synonyms task using the ukWaC- Corpus. The first column gives the results of ranking the pairs according to their similarity. The second column shows the dependency of the similarity of the word pairs on their frequency expressed the R^2 value of the moving average. The last column gives the results when the pairs are ranked according to the residual values.

Method	AUC (sim)	R^2	AUC (res)
plain_cos	0.66	0.22	0.77
plain_jsd	0.43	0.86	0.72
plain_pmi	0.67	0.33	0.85
svd	0.81	0.04	0.85
ri_cos	0.60	0.28	0.72
ri_jsd	0.41	0.94	0.70

Table 4: Results of 6 different distributional similarity methods on ranking 1953 pairs of synonyms and non-synonyms from which at least one word occurs less than 100 times in the ukWaC-Corpus. The first column gives the results of ranking the pairs according to their similarity. The second column gives the results when the pairs are ranked according to the residual values.

Method	AUC (sim)	AUC (res)
plain_cos	0.65	0.71
plain_jsd	0.53	0.64
plain_pmi	0.73	0.81
svd	0.80	0.82
ri_cos	0.59	0.65
ri_jsd	0.50	0.61

quency bias is better than the results on the complete data set. This is as expected, since the frequency range is clearly reduced in this subset. When we rank the pairs using the residual values, the results of all methods stay behind those on the complete data set.

6 Discussion

We clearly see that all methods become better when more data are available. However, all methods have the potential to make good predictions for less frequent words. The method using SVD is only slightly worse on the less frequent data. Thus we see that the best methods still give useful results for infrequent words, contradicting the findings of Ferret (2010).

For the cosine similarity and the JSD the dependency on the word frequency can intuitively be understood as follows. The cosine depends only on the dimensions for which both vectors have a non-zero value. If the vectors become less sparse, since we have seen more different contexts, it is not surprising that the cosine tends to become larger. The JSD also depends only on the dimensions for which both vectors have a non-zero value. This can be seen if we rewrite the JSD for two probability density functions p and q as

$$\begin{aligned}\mathrm{JSD}(p,q) &= \tfrac{1}{2}D(p||\tfrac{1}{2}p+\tfrac{1}{2}q)+\tfrac{1}{2}D(q||\tfrac{1}{2}p+\tfrac{1}{2}q)\\ &= \log 2+\tfrac{1}{2}\sum_{t:p(t)\neq 0\,\wedge\, q(t)\neq 0}\Big(p(t)\log\big(\tfrac{p(t)}{p(t)+q(t)}\big)\\ &+ q(t)\log\big(\tfrac{q(t)}{p(t)+q(t)}\big)\Big), \qquad (2)\end{aligned}$$

where $D(p,q)$ is the Kullback-Liebler divergence of p and q. The differences between cosine and JSD cannot be explained that easily. If we weight the features using PPMI the influence of words just occurring a few times in the context of a word is reduced. Thus the similarity caused by irrelevant words just randomly occurring in the context of both words when we consider enough data, is reduced. The influence of irrelevant features is further reduced when SVD is used.

Furthermore we see that the dependency on the frequency for the methods using random indexing is larger than for the corresponding plain methods. For random indexing we included much more (infrequent) context words as features. Thus there are more factors that potentially cause the differences.

The data set of the ranking task was first used by Wartena (2013a). When we compare the results with the results presented there, we see that we get exactly the same result for *ri_jsd*, though the configuration is somewhat different: we use 10 000 dimensions and a window of 4 words, whereas Wartena (2013a) used 20 000 dimensions and used all words in the sentence as features. For the *ri_cos* method the results are worse than those presented there. Wartena (2013a) also gives a ranking by the residual values. The results given there are much better than those found here and even slightly better than those found using SVD. The difference between the both studies is, that the modeling in Wartena (2013a) is not based on

the frequency of the words but on the number of non-zero values in the feature vectors.

Of course, it would be easy to obtain better results, by using other additional features for the ranking. E.g. the synonyms in the data set tend to have a lower frequency than the unrelated word pairs. Moreover, many synonyms are just spelling variants, that could be detected easily using edit distance or bigram overlap.

7 Conclusions and Future Work

Though the dependency of word similarities in distributional models on their frequencies is already known since a decade, the issue has received little attention. In the present paper we investigated the influence of word frequency on 6 different methods to compute distributional similarity. Thus the paper extends previous work in which only random indexing was considered or in which a frequency bias was observed for various methods but in which the correlation between frequency and similarity was not investigated in more detail.

We find that all tested methods except the one using SVD for dimensionality reduction are strongly dependent on frequency of the words. We find the dependency as well for cosine similarity as for Jensen-Shannon divergence. The dependencies are found consistently on two different data sets. The second data set consist of pairs of synonyms and unrelated words. We have shown that the methods that are strongly dependent on word frequency nevertheless have the potential to discriminate between pairs of synonyms and unrelated words, when we do not use the absolute similarity but the similarity relative to the similarity expected on the base of the word frequency.

The superiority of the method using point wise mutual information for feature weighting, SVD for dimensionality reduction and the cosine as similarity measure for feature vectors was already found in a number of other studies. However, the present study reveals one of the factors that are responsible for the performance differences: the distortion by the word frequencies.

We now could conclude that we know which method to use. However, SVD is computationally demanding and not feasible in all situations. The fact that we have shown that other methods can give similar results when we correct for the frequency bias, encourages us to search for similarity measures and feature weighting schemes that are less sensitive for word frequency. A different direction that we will pursue is smoothing of the feature vectors of infrequent words in order to compensate for the effects of a low number of observations.

References

Marco Baroni, S. Bernardini, A. Ferraresi, and E. Zanchetta. 2009. The WaCky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora. *Language Resources and Evaluation 43 (3): 209-226*, 43(3):209–226.

John A. Bullinaria and Joseph P. Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. (39):510–526.

John A. Bullinaria and Joseph P. Levy. 2012. Extracting Semantic Representations from Word Co-occurrence Statistics: Stop-lists, Stemming and SVD. (44):890–907.

Carolyn J. Crouch. 1990. An approach to the automatic construction of global thesauri. *Information Processing & Management*, 5:629–640.

James R. Curran and Marc Moens. 2002. Improvements in Automatic Thesaurus Extraction. In *Unsupervised Lexical Acquisition: Proceedings of the Workshop of the ACL Special Interest Group on the Lexicon (SIGLAX)*, pages 59–66.

Olivier Ferret. 2010. Testing Semantic Similarity Measures for Extracting Synonyms from a Corpus. In Guy de Pauw, editor, *LREC 2010, Seventh International Conference on Language Resources and Evaluation*, pages 3338–3343.

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 the 10th International Conference on World Wide Web*, WWW '01, pages 406–414, New York and NY and USA. ACM.

Dayne Freitag, Matthias Blume, John Byrnes, Edmond Chow, Sadik Kapadia, Richard Rohwer, and Zhiqiang Wang. 2005. New Experiments in Distributional Representations of Synonymy. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, CONLL '05, pages 25–32, Stroudsburg and PA and USA. Association for Computational Linguistics.

Eugenie Giesbrecht. 2010. Towards a Matrix-based Distributional Model of meaning. In *Proceedings*

of the NAACL HLT 2010 Student Research Workshop, pages 23–28, Los Angeles, California. ACL.

Jussi Karlgren and Magnus Sahlgren. 2001. From Words to Understanding. In *Foundations of Real-World Intelligence*, pages 294–308. CSLI Publications, Stanford and California.

Jun'ichi Kazama, Stijn de Saeger, Kow Kuroda, Masaki Murata, and Kentaro Torisawa. 2010. A Baysesian Method for Robust Estimation of Distributional Similarities. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 247–356.

Douwe Kiela and Stephen Clark. 2014. A Systematic Study of Semantic Vector Space Model Parameters. In *2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*.

Ioannis Korkontzelos, Torsten Zesch, Fabio Massimo Zanzotto, and Chris Biemann. 2013. SemEval-2013 Task 5: Evaluating Phrasal Semantics. In *Proceedings of the 7th International Workshop on Semantic Evaluation (Semeval 2013)*.

Thomas K. Landauer and Susan T Dumais. 1997. A solution to Plato's problem: the latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240.

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

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive science*, 34(8):1388–1429.

Malti Patel, John A. Bullinaria, and Joseph P. Levy. 1998. Extracting semantic representations from large text corpora. In *4th Neural Computation and Psychology Workshop, London, 9–11 April 1997*, pages 199–212.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual Correlates of Synonymy. *Commun. ACM*, 8(10):627–633.

Helmut Schmid. 1995. Improvements in Part-of-Speech Tagging with an Application to German. In *Proceedings of the ACL SIGDAT-Workshop*.

Peter D. Turney and Patrick Pantel. 2010. From Frequency to Meaning: Vector Space Models of Semantics. *Journal of Artificial Intelligence Research*, (37):141–188.

Christian Wartena. 2013a. Distributional similarity of words with different frequencies. In *Proceedings of the 13th Dutch-Belgian Workshop on Information Retrieval*.

Christian Wartena. 2013b. HsH: Estimating semantic similarity of words and short phrases with frequency normalized distance measures. In *Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), Atlanta, Georgia, USA*.

Julie Weeds, David J. Weir, and Diana McCarthy. 2004. Characterising Measures of Lexical Distributional Similarity. In *COLING 2004, Proceedings of the 20th International Conference on Computational Linguistics*.

Maayan Zhitomirsky-Geffet and Ido Dagan. 2009. Bootstrapping Distributional Feature Vector Quality. *Computational Linguistics*, 25(3):435–461.