Second Order Co-occurrence PMI for Determining the Semantic Similarity of Words

Md. Aminul Islam and Diana Inkpen
School of Information Technology and Engineering
University of Ottawa
Ottawa, Ontario, Canada, K1N 6N5
{mdislam,diana}@site.uottawa.ca

Abstract

This paper presents a new corpus-based method for calculating the semantic similarity of two target words. Our method, called Second Order Co-occurrence PMI (SOC-PMI), uses Pointwise Mutual Information to sort lists of important neighbor words of the two target words. Then we consider the words which are common in both lists and aggregate their PMI values (from the opposite list) to calculate the relative semantic similarity. Our method was empirically evaluated using Miller and Charler’s (1991) 30 noun pair subset, Rubenstein and Goodenough’s (1965) 65 noun pairs, 80 synonym test questions from the Test of English as a Foreign Language (TOEFL), and 50 synonym test questions from a collection of English as a Second Language (ESL) tests. Evaluation results show that our method outperforms several competing corpus-based methods.

1. Introduction

Semantic relatedness refers to the degree to which two concepts or words are related (or not) whereas semantic similarity is a special case or a subset of semantic relatedness. Humans are able to easily judge if a pair of words are related in some way. For example, most would agree that apple and orange are more related than are apple and toothbrush. Budanitsky and Hirst (2004) point out that semantic similarity is used when similar entities such as apple and orange or table and furniture are compared. These entities are close to each other in an is-a hierarchy. For example, apple and orange are hyponyms of fruit and table is a hyponym of furniture. However, even dissimilar entities may be semantically related, for example, glass and water, tree and shade, or gym and weights. In this case the two entities are intrinsically not similar, but are related by some relationship. Sometimes this relationship may be one of the classical relationships such as meronymy (is part of) as in computer – keyboard or a non-classical one as in glass - water, tree – shade and gym – weights. Thus two entities are semantically related if they are semantically similar (close together in the is-a hierarchy) or share any other classical or non-classical relationships.

Measures of the semantic similarity of words have been used for a long time in applications in natural language processing and related areas, such as the automatic creation of thesauri (Grefenstette, 1993; Lin, 1998; Li and Abe, 1998), automatic indexing, text annotation and summarization (Lin and Hovy, 2003), text classification, word sense disambiguation (Lesk, 1986; Yarowsky, 1992; Li and Abe, 1998), information extraction and retrieval (Buckley et al., 1995; Vechtomova and Robertson, 2000; Xu and Croft, 2000), lexical selection, automatic correction of word errors in text and discovering word senses directly from text (Pantel and Lin, 2002).

A word similarity measure is also used for language modeling by grouping similar words into classes (Brown et al., 1992). In databases, word similarity can be used to solve semantic heterogeneity, a key problem in any data sharing system whether it is a federated database, a data integration system, a message passing system, a web service, or a peer-to-peer data management system (Madhavan et al., 2005).

This paper is organized as follow: Section 2 presents an overview of the related work. Our SOC-PMI word similarity method is described in Section 3. A walkthrough example of the method is presented in Section 4. Evaluation and experimental results are discussed in Section 5. Section 6 discusses the potential applications of SOC-PMI and we conclude in Section 7.

2. Related Work

Many different measures of semantic similarity between word pairs have been proposed, some using statistical or distributional techniques (Grefenstette, 1992; Lin, 1998), some using lexical databases (thesaurus), and some hybrid approaches, combining distributional and lexical techniques. PMI-IR (Turney, 2001) is a statistical approach that uses a huge data source: the web. Another well-known statistical approach to measuring semantic similarity is Latent Semantic Analysis (LSA) (Landauer and Dumais, 1997). We will briefly discuss these two approaches in next sub sections.

Individual words in a given a text corpus have more or less differing contexts around them. The context of a word is composed of words co-occurring with it within a certain window around it. Distributional measures use statistics acquired from a large text corpora to determine how similar the contexts of two words are. These measures are also used as approximations to measures of semantic similarity of words, because words found in similar contexts tend to be semantically similar. Such measures have traditionally been referred to as measures of distributional similarity. If two words have many co-occurring words, then similar things are being said about both of them and therefore they are likely to be semantically similar. Conversely, if two words are semantically similar then they are likely to be used in a similar fashion in text and thus end up with many common co-occurrences. For example, the semantically similar car and vehicle are expected to have a number of common co-
occurring words such as parking, garage, model, industry, accident, traffic, and so on, in a large enough text corpus.

Various distributional similarity measures were discussed in (Weeds et al., 2004) where co-occurrence types of a target word are the contexts in which it occurs and these have associated frequencies which may be used to form probability estimates. Lesk (1969) was one of the first to apply the cosine measure, which computes the cosine of the angle between two vectors, to word similarity. The Jensen-Shannon (JS) divergence measure (Rao, 1983; Dagan et al., 1999) and the skew divergence measure (Lee, 1999) are based on the Kullback-Leibler (KL) divergence measure. Jaccard’s coefficient (Salton and McGill, 1983) calculates the proportion of features belonging to either word that are shared by both words. In the simplest case, the features of a word are defined as the contexts in which it has been seen to occur. Pointwise Mutual Information (PMI) was first used to measure word similarity by (Church and Hanks 1990) where positive values indicate that words occur together more than would be expected under an independence assumption and negative values indicate that one word tends to appear only when the other does not. Jaccard-MI is a variant (Lin, 1998) in which the features of a word are those contexts for which the pointwise mutual information between the word and the context is positive. Average Mutual Information corresponds to the expected value of two random variables using the same equation as PMI and was used as a word similarity measure by (Rosenfeld, 1996; Dagan et al., 1999). Cosine of pointwise mutual information was used by (Pantel and Lin, 2002) to uncover word senses from text. $L_1$ norm method was proposed as an alternative word similarity measure in language modeling to overcome zero-frequency problems for bigrams (Dagan et al., 1999). A likelihood ratio was used by (Dunning, 1993) to test word similarity under the assumption that the words in text have a binomial distribution.

2.1. Latent Semantic Analysis (LSA)

LSA (Landauer and Dumais, 1997), a high-dimensional linear association model, analyzes a large corpus of natural text and generate a representation that captures the similarity of words and text passages. The underlying idea is that the aggregation of all the word contexts in which a given word does and does not appear provides a set of mutual constraints that largely determines the similarity of meaning of words and sets of words to each other (Landauer et al., 1998). The model tries to answer how people acquire as much knowledge as they do on the basis of as little information as they get. It uses the Singular Value Decomposition (SVD) to find the semantic representations of words by analyzing the statistical relationships among words in a large corpus of text. The corpus is broken up into chunks of texts approximately the size of a text or paragraph or small document. Analyzing each text or paragraph, the number of occurrences of each word is set in a matrix with a column for each word and a row for each paragraph. Then each cell of the matrix (a word by context matrix, $X$), is transformed from the raw frequency count into the log of the count. After that each cell is divided by the entropy of the column, given by $-\sum p \log p$, where the summation is over all the paragraphs the word appeared.

The next step is to apply SVD to $X$, to decompose $X$ into a product of three matrices

$$X = WSP^\top,$$

where, $W$ and $P$ are in column orthonormal form (i.e., columns are orthogonal) and $S$ is the diagonal matrix of non-zero entries (singular values). To reduce dimensions, the rows of $W$ and $P$ corresponding to the highest entries of $S$ are kept. In other words, the new lower-dimensional matrices $W_t$, $P_t$ and $S_t$ are the matrices produced by removing the columns and rows with smallest singular values from $W$, $P$ and $S$. This new matrix

$$X_t = W_t S_t P_t^\top$$

is a compressed matrix which represents all the words and text samples in a lower dimensional space. Then the similarity of two words, using LSA, is measured by the cosine of the angle between their corresponding row vectors.

2.2. PMI-IR

PMI-IR (Turney, 2001), a simple unsupervised learning algorithm for recognizing synonyms, uses Pointwise Mutual Information as follows:

$$\text{score}_t(\text{choice}) = \frac{p(\text{problem} \& \text{choice})}{p(\text{choice})}.$$  

Here, $p(\text{problem})$ is the probability that the problem word and choice appear. In other words, each choice is simply scored by the conditional probability of the problem word, given the choice word, $p(\text{problem} | \text{choice})$. If problem and choice are statistically independent, then the probability that they co-occur is given by the product $p(\text{problem}) \cdot p(\text{choice})$. If they are not independent, and there is a tendency to co-occur, then $p(\text{problem} | \text{choice})$ will be greater than $p(\text{problem})$. PMI-IR uses AltaVista Advanced Search query syntax to calculate the probabilities. In the simplest case, two words co-occur when they appear in the same document:

$$\text{score}_t(\text{choice}) = \frac{\text{hits}_t(\text{problem AND choice})}{\text{hits}_t(\text{choice})}.$$  

Here, hits($\text{choice}$) is the number of hits (the number of documents retrieved) when the query $\text{choice}$ is given to AltaVista. AltaVista provides how many documents contain both problem and choice, and then how many documents contain choice, alone. The ratio of these two numbers is the score for choice. There are three other versions of this scoring equation based on the closeness of the pairs in documents, considering antonyms, and taking context into account.

3. Second Order Co-occurrence PMI Method

Let $W_1$ and $W_2$ be the two words for which we need to determine the semantic similarity and $C = \{c_1, c_2, \ldots, c_m\}$ denotes a large corpus of text (after some preprocessing e.g., stop words elimination and lemmatization) containing $m$ words (tokens). Also, let $T = \{t_1, t_2, \ldots, t_n\}$ be the set of all unique words (types) which occur in the corpus. Unlike the corpus $C$, which is an ordered list containing many occurrences of the same words, $T$ is a set containing no repeated words. Throughout this section, we will use $W$ to denote either $W_1$ or $W_2$.

We set a parameter $\alpha$, which determines how many words before and after the target word $W$, will be included in the context window. The window also contains the
target word $W$ itself, resulting in a window size of $2\alpha + 1$ words. The steps in determining the semantic similarity involve scanning the corpus and then extracting some functions related to frequency counts.

We define the type frequency function,
\[ f(t_i) = \{k: c_k = t_i\}, \]
where $i = 1, 2, \ldots, n$ which tells us how many times the type $t_i$ appeared in the entire corpus. Let
\[ f(t_i, W) = \{k: t_i = W \text{ and } c_k = t_i\}, \]
where $i = 1, 2, \ldots, n$ and $-\alpha \leq j \leq \alpha$, be the bigram frequency function. $f(t_i, W)$ tells us how many times word $t_i$ appeared with word $W$ in a window of size $2\alpha + 1$ words.

Then we define pointwise mutual information function for only those words having $f(t_i, W) > 0$,
\[ f_{pmi}(t_i, W) = \log \frac{f(t_i, W)}{f(t_i) f(W)}, \]
where $f(t_i) f(W) > 0$ and $m$ is total number of tokens in corpus $C$ as mentioned earlier. Now, for word $W_1$, we define a set of words, $X$, sorted in descending order by their PMI values with $W_1$ and taken the top-most $\beta_1$ words having $f_{pmi}(t_i, W_1) > 0$.

\[ X = \{X_i\}, \text{ where } i = 1, 2, \ldots, \beta_1 \]

and $f_{pmi}(t_i, W_1) \geq f_{pmi}(t_2, W_1) \geq \ldots \geq f_{pmi}(t_{\beta_1}, W_1) \geq f_{pmi}(t_2, W_1)$

Similarly, for word $W_2$, we define a set of words, $Y$, sorted in descending order by their PMI values with $W_2$ and taken the top-most $\beta_2$ words having $f_{pmi}(t_i, W_2) > 0$.

\[ Y = \{Y_i\}, \text{ where } i = 1, 2, \ldots, \beta_2 \]

and $f_{pmi}(t_i, W_2) \geq f_{pmi}(t_2, W_2) \geq \ldots \geq f_{pmi}(t_{\beta_2}, W_2) \geq f_{pmi}(t_2, W_2)$

Note that we have not yet determined the value for $\beta$'s (either $\beta_1$ or $\beta_2$) which actually depend on the word $W$ and the number of types in the corpus (this will be discussed in the next section).

Again, we define the $\beta$-PMI summation function. For word $W_1$, the $\beta$-PMI summation function is:
\[ f_{\beta}(W_1) = \frac{\beta_1}{\beta_2} \sum_{i=1}^{\beta_1} f_{pmi}(X_i, W_2) \]

where, $f_{pmi}(X_i, W_2) > 0$ and $f_{pmi}(X_i, W_2) > 0$

which sums all the positive PMI values of words in the set $Y$ also common to the words in the set $X$. In other words, this function actually aggregates the positive PMI values of all the semantically close words of $W_2$ which are also common in $W_1$. Note that we call it semantically-close because all these words have high PMI values with $W_2$ and this doesn’t ensure the closeness with respect to the distance within the window size.

Similarly, for word $W_2$, the $\beta$-PMI summation function is:
\[ f_{\beta}(W_2) = \frac{\beta_2}{\beta_1} \sum_{j=1}^{\beta_2} f_{pmi}(Y_j, W_1) \]

where, $f_{pmi}(Y_j, W_1) > 0$ and $f_{pmi}(Y_j, W_1) > 0$

which sums all the positive PMI values of words in the set $X$ also common to the words in the set $Y$. In other words, this function aggregates the positive PMI values of all the semantically-close words of $W_1$ which are also common in $W_2$. We have not discussed the criteria for choosing the exponential parameter $\gamma$ (this will be discussed in the next subsection).

Finally, we define the semantic PMI similarity function between two words, $W_1$ and $W_2$,
\[ \text{Sim}(W_1, W_2) = \frac{f_{\beta}(W_1)}{\beta_1} + \frac{f_{\beta}(W_2)}{\beta_2} \]

3.1. Choosing the Values of $\beta$ and $\gamma$

The value of $\beta$ is related to how many times the word, $W$ appears in the corpus, i.e., the frequency of $W$ as well as the number of types in the corpus. We define $\beta$ as
\[ \beta = \{\log (f(W_i))\}^{\beta^2} (\log_2(|n|))^{\delta}, \]
where $\delta$ is a constant and for all of our experiments we used $\delta = 0.5$. The value of $\delta$ depends on the size of the corpus. The smaller the corpus we use, the smaller the value of $\delta$ we should choose. If we lower the value of $\beta$ we lose some important / interesting words, and if we increase it we consider more words common to both $W_1$ and $W_2$ and this significantly degrades the result.

$\gamma$ should have a value greater than 1. The higher we choose the value of $\gamma$, the greater emphasis on words having very high PMI values with $W$. For all our experiments, we chose $\gamma = 3$.

4. A Walk-Through Example

Suppose we want to determine the semantic similarity between words $W_1 = \text{car}$ and $W_2 = \text{automobile}$ and the following 12 sentences (Table 1) are our corpus of text after preprocessing (stop-words elimination and lemmatization). Here, we have tokens $m = 70$ and types $n = 43$ (the types and the corresponding frequencies are in Table 2). The bigram frequencies for word $W_1$ and $W_2$ in a window of 11 words ($\alpha = 5$) are in Table 3.

For this small corpus, we have chosen $\delta = 0.7$ and $\gamma = 3$. Here, $\beta_1 = (\log (f(W_i)))^{\beta^2} (\log_2(|n|))^{24.88}$, and similarly, $\beta_2 = 24.88$

Now we determine the set $X$ of words sorted in descending order by their PMI values with $W_1$ and take the top most 23 ($\beta_1$) (it could have been maximum 24) words (see Table 3) having $f_{pmi}(t_i, W_1) > 0$. Similarly, the number of words we got in $Y$ is 19 ($\beta_2$), though it could have been at most 24.

| 1 | pursuit accident claim car driver exclude |
| 2 | soak motorist company car driver risky |
| 3 | company car driver tend travel farther |
| 4 | job engineer disappear fall mechanical engineer car industry worst affect |
| 5 | sign recession car industry |
| 6 | brightest engineer moment car industry |
| 7 | yugoslavia benefit direct investment automobile industry |
| 8 | acreage expand emergence automobile industry |
| 9 | automobile industry among hardest hit recession |
| 10 | automobile industry largely male force |
| 11 | component supplier automobile industry expand |
| 12 | client industry manufacturer component automobile industry |

Table 1: Sample texts after cleaning.

\(^1\) Actually we are using a large corpus (the BNC), but we use this very small number of texts in this example to explain the method.
Then we compute:
\[
f^\beta(W_1) = (f_{pmi}("recession", W_2))^\gamma + (f_{pmi}("industry", W_2))^\gamma = (2.544)^\gamma + (3.029)^\gamma = 44.255
\]
Similarly,
\[
f^\beta(W_2) = (f_{pmi}("industry", W_1))^\gamma + (f_{pmi}("recession", W_1))^\gamma = (1.807)^\gamma + (2.544)^\gamma = 22.364
\]
Therefore,
\[
Sim(W_1, W_2) = \frac{f^\beta(W_1) \cdot f^\beta(W_2)}{\frac{1}{\gamma_1} + \frac{1}{\gamma_2}} = \frac{44.255 \cdot 22.364}{\frac{1}{1} + \frac{1}{1}} = 3.101
\]

5. Experimental Results

Our method was empirically evaluated on the task of solving 80 synonym TOEFL questions and 50 synonym ESL questions; and using Miller and Charles’ (1991) 30 noun pairs subset and Rubenstein and Goodenough’s (1965) 65 noun pairs.

We computed the SOC-PMI similarity values using the BNC as a source of frequencies and contexts. The size of this corpus is approximately 100 million words, and it is a balanced corpus: it contains texts from various sources, general British English.

Landauer and Dumais (1997) employed word similarity measures to answer 80 synonym test questions from the Test of English as a Foreign Language (TOEFL) using Latent Semantic Analysis (LSA). Turney (2001) applied his Pointwise Mutual Information and Information Retrieval (PMI-IR) measure to answer 50 synonym test questions from a collection of English as a Second Language (ESL) tests and the same 80 TOEFL questions set that Landauer and Dumais (1997) used.

For the 80 TOEFL questions, the SOC-PMI method correctly answered 76.25% of the questions, as shown in Table 4. This is an improvement over the results presented by Landauer and Dumais (1997), using LSA, where 64.5% of the questions were answered correctly, and Turney (2001), using the PMI-IR algorithm, where the best result was 73.75%. A human average score on the same question set is 64.5% (Landauer and Dumais, 1997).

| Method name | Number of correct test answers | Question or answer words not found | Percentage of correct answers |
|-------------|-------------------------------|-----------------------------------|-------------------------------|
| Penguin Roget | 63 | 26 | 78.75% |
| SOC-PMI | 61 | 4 | 76.25% |
| PMI-IR | 59 | 0 | 73.75% |
| LSA | 51.5 | 0 | 64.37% |
| Lin | 32 | 42 | 40.00% |

Table 4: Results on the 80 TOEFL questions

\[http://www.natcorp.ox.ac.uk/\]
First, we need to sort out the significant words list based on PMI values for the word (say, \( x \)). We are interested to find the synonyms. If there are \( n \) significant words in this words list, we will apply the SOC-PMI method for each possible pair mapping from \( x \) to \( n \). Instead of taking the similarity value, we will consider all the second order co-occurrence types and sort out this types list based on PMI values. The words top on the list could be the best candidates for synonyms of the word. This could be a future addition to our proposed method.

Detecting semantic outliers in speech recognition transcripts can use semantic similarity measures (Inkpen and Desilets, 2005) and a corpus-based similarity measures played an important role because of its large type coverage. The corpus-based measure was shown to perform better than the Roget-based measure in the task.

### 7. Conclusion

In this paper, we evaluate the new similarity measure and compare it with existing similarity measures. We performed intrinsic evaluation on the noun pairs mentioned above. We also performed a task-based evaluation: solving synonyms test questions. We plan to apply our method to other tasks, such as measuring the semantic similarity of two texts and detecting semantic outliers in speech recognition transcripts. One of the main characteristics of the SOC-PMI method is that we can determine the semantic similarity of two words even though they do not co-occur within the window size at all in the corpus. Actually, we are considering the second order co-occurrences, as we are judging by the co-occurrences of the neighbor words, not only the co-occurrence of the two target words. This is not the case for PMI-IR and many other corpus-based semantic similarity measures.

### Acknowledgements

We want to thank Thomas K. Landauer, Department of Psychology, University of Colorado, for providing the 80 TOEFL questions. We are also grateful to Peter D. Turney, National Research Council of Canada, who provided the 50 ESL questions.

### References

Brown, P. F., DeSouza, P. V., Mercer, R. L., Watson, T. J., Della Pietra, V. J. and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18:467-479.

Budanitsky, Alexander and Hirst, Graeme. (2006). Evaluating WordNet-based measures of semantic distance. *Computational Linguistics*, 32(1).

Buckley, C., Salton, J. A. and Singhal, A. (1995). Automatic query expansion using Smart: TREC 3. In *The third Text Retrieval Conference*, Gaithersburg, MD.

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

Dagan, I., Lee, L. and Pereira, F.C.N. (1999). Similarity based models of word coocurrence probabilities. *Machine Learning*, 34(1-3):43-69.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19:61-74.
Grefenstette, G. (1993). Automatic thesaurus generation from raw text using knowledge-poor techniques. In Making sense of Words, 9th Annual Conference of the UW Centre for the New OED and Text Research.

Grefenstette, G. (1992). Finding Semantic Similarity in Raw Text: The Deese Antonyms. In: R. Goldman, P. Norvig, E. Charniak and B. Gale (eds.), Working Notes of the AAAI Fall Symposium on Probabilistic Approaches to Natural Language. AAAI Press, pp. 61-65.

Inkpen, Diana and Désilets, Alain. (2005). Semantic Similarity for Detecting Recognition Errors in Automatic Speech Transcripts. EMNLP 2005, Vancouver, Canada.

Jarmasz, M. and Szpakowicz, S. (2003). Roget's thesaurus and semantic similarity, International Conference RANLP-2003, Borovets, Bulgaria, pp. 212-219.

Landauer, T. K. and Dumais, S. T. (1997). A Solution to Plato’s Problem: The Latent Semantic Analysis Theory of the Acquisition, Induction, and Representation of Knowledge. Psychological Review, 104(2):211-240.

Lee, L. (1999). Measures of distributional similarity. In Proceedings of ACL-1999, pp. 23-32.

Lesk, M. E. (1969). Word-word associations in document retrieval systems. American Documentation, 20(1):27-38.

Lesk, M. E. (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In Proceedings of the SIGDOC Conference, Toronto.

Li, H. and Abe, N. (1998). Word clustering and disambiguation based on co-occurrence data. In COLING-ACL, pp. 749-755.

Lin, C. Y. and Hovy, E. H. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In Proceedings of the Human Language Technology Conference (HLT-NAACL 2003), Edmonton, Canada.

Lin, D. (1998). Automatic retrieval and clustering of similar words. In COLING-ACL, pp. 768-774.

Madhavan, J., Bernstein, P., Doan, A. and Halevy, A. (2005). Corpus-based Schema Matching. In International Conference on Data Engineering (ICDE-05).

Miller, G. A. and Charles, W. G. (1991). Contextual correlates of semantic similarity. Language and Cognitive Processes, 6(1):1-28.

Pantel, P. and Lin, D. (2002). Discovering word senses from text. In Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 613-619.

Pedersen, T., Patwardhan, S. and Michelizzi, J. (2004). WordNet::Similarity - Measuring the Relatedness of Concepts, in Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI-04), July 25-29, San Jose, CA (Intelligent Systems Demonstration).

Rao, C. R. (1983). Diversity: Its measurement, decomposition, apportionment and analysis. Sankhya: Indian Journal of Statistics, 44(A):1-22.

Resnik, P. (1995). Using information content to evaluate semantic similarity. In Proceedings of 14th International Joint Conference on Artificial Intelligence, Montreal, pp. 448-453.

Rosenfeld, R. (1996). A maximum entropy approach to adaptive statistical language modeling. Computer speech and language. 10:187-228.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual correlates of synonymy. Communications of the ACM, 8(10): 627-633.

Salton, G. and McGill, M.J. (1983). Introduction to Modern Information Retrieval. McGraw-Hill.

Turney, P. D. (2001). Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In Proceedings of the Twelfth European Conference on Machine Learning (ECML-2001), pp. 491-502.

Vechtomova, O. and Robertson, S. (2000). Integration of collocation statistics into the probabilistic retrieval model. In 22nd Annual Colloquium on Information Retrieval Research, Cambridge, England.

Weeds, J., Weir, D. and McCarthy, D. (2004). Characterising measures of lexical distributional similarity. In Proceedings of the 20th International Conference of Computational Linguistics, COLING-2004, Geneva, Switzerland, pp. 1015-1021.

Xu, J. and Croft, B. (2000). Improving the effectiveness of information retrieval. ACM Transactions on Information Systems, 18(1):79-112.

Yarowsky, D. (1992). Word-sense disambiguation using statistical models of Roget’s categories trained on large corpora. In Proceedings of COLING-92, Nantes, France, pp. 454-460.