JISTaP http://www.jistap.org
Journal of Information Science Theory and Practice
eISSN : 2287-4577 pISSN : 2287-9099

Research Paper
J Inf Sci Theory Pract 8(2): 06-17, 2020
https://doi.org/10.1633/JISTaP.2020.8.2.1

Empirical Comparison of Word Similarity Measures Based on Co-Occurrence, Context, and a Vector Space Model

Natsuki Kadowaki

Graduate School of Library and Information Science,
Keio University, Tokyo, Japan
E-mail: kadowaki.72@keio.jp

Kazuaki Kishida*

Faculty of Letters, Keio University, Tokyo, Japan
E-mail: kz_kishida@keio.jp

ABSTRACT

Word similarity is often measured to enhance system performance in the information retrieval field and other related areas. This paper reports on an experimental comparison of values for word similarity measures that were computed based on 50 intentionally selected words from a Reuters corpus. There were three targets, including (1) co-occurrence-based similarity measures (for which a co-occurrence frequency is counted as the number of documents or sentences), (2) context-based distributional similarity measures obtained from a latent Dirichlet allocation (LDA), nonnegative matrix factorization (NMF), and Word2Vec algorithm, and (3) similarity measures computed from the tf-idf weights of each word according to a vector space model (VSM). Here, a Pearson correlation coefficient for a pair of VSM-based similarity measures and co-occurrence-based similarity measures according to the number of documents was highest. Group-average agglomerative hierarchical clustering was also applied to similarity matrices computed by individual measures. An evaluation of the cluster sets according to an answer set revealed that VSM- and LDA-based similarity measures performed best.

Keywords: word similarity, word clustering, topic model, word embedding

Open Access

Received date: January 23, 2020
Accepted date: April 1, 2020

***Corresponding Author:** Kazuaki Kishida
Professor
Faculty of Letters, Keio University, 2-15-45 Mita, Minato-ku,
Tokyo 108-8345, Japan
E-mail: kz_kishida@keio.jp

1. INTRODUCTION

Word similarity or semantic similarity between words is often determined to improve the effectiveness of some applications in the field of information retrieval (IR) and other related areas, such as text categorization. For example, suppose that a very short query is used to search a database and thus returns insufficient or irrelevant results. This is because the query did not contain words that accurately represented the user's needs. However, a new query with added words that are similar to the original query words should obtain better results, including more relevant documents that only contain the newly added words. This is referred to as a query expansion technique (Zazo, Figuerola, Berrocal, & Rodríguez, 2005). Word similarity measurements also play important roles in bibliometric studies that attempt to articulate an "intellectual structure" inherent to a set of scientific documents (i.e., a co-word analysis) (e.g., Khasseh, Soheili, Moghaddam, & Chelak, 2017; Ravikumar, Agrahari, & Singh, 2015).

Although human generated thesauri such as WordNet are sometimes used to find similar words, such a task is usually accomplished through automatic corpus processing. In this context, word similarity can be determined in the three following sources: 1) co-occurrence frequencies of two words, 2) the degree to which context words appear around two corresponding target words, and 3) word vectors consisting of weights in individual documents. First, co-occurrence frequencies are typically counted as the number of sentences or documents in which both of two target words appear. Second, a context word information method entails that a feature vector for each word is algorithmically estimated. Here, the Word2Vec algorithm (Mikolov, Yih, & Zweig, 2013) has been widely applied to estimate feature vectors. In comparing these methods, Liebeskind, Dagan, and Schler (2018) called the first "a first-order, co-occurrence-based approach," but referred to the second as "a second-order, distributional similarity approach" (p. 1446). Third, some studies have constructed word vectors by juxtaposing the tf-idf weighs of each word in individual documents according to a vector space model (VSM) in the IR field. Here, cosine values of the word vectors are computed as similarity measures.

Researchers have variously used the above three methods to conduct word similarity measures depending on their specific applications and needs. However, it is unclear which similarity measure is better. This is because few studies have systematically compared results between the three. For example, although Dagan, Lee, and Pereira (1999) tried to examine performance of some similarity measures through an experiment on word sense disambiguation, it did not cover all three types of measures mentioned above. In particular, because Word2Vec is a new algorithm, its effect on computation of word similarity has not yet been known enough, and so it would be worthwhile to compare word similarity computed by Word2Vec with those by traditional methods such as co-occurrence-based and VSM-based approaches. As a result, empirical comparison among the three types of word similarity measure using a common dataset would bring us a new insight on the measures.

Thus, this study empirically investigated results from the above three-word similarity measures through a comparative experiment that implemented a portion of the Reuters Corpus Volume 1 (RCV1) as a test set. More specifically, 50 words were carefully selected as a sample. Similarity matrices were then calculated for the sample using each of the three methods. Here, the purpose was to directly compare similarity values through a Pearson correlation coefficient and evaluate the clustering results obtained by applying a standard hierarchical clustering algorithm to the matrices. This provided new insight on the effectiveness of word similarity measures as a metric for constructing word clusters for use in many applications.

The rest of this paper proceeds as follows: Section 2 reviews the three abovementioned types of word similarity measures and discusses previous related studies, while Section 3 describes this study's experimental method of empirically comparing each measure, and Section 4 discusses the results.

2. COMPUTING WORD SIMILARITY

This section reviews three types of word similarity measures and discusses related studies.

2.1. Co-Occurrence-Based Similarity

The number of documents in which word w_j appears (i.e., document frequency) is denoted by n_j. If n_{jk} indicates the number of documents including both w_j and w_k, then the degree of similarity between them (which is written as s_{jk}) can be computed as follows:

$$s_{jk} = \frac{n_{jk}}{\sqrt{n_j n_k}} \tag{1}$$

which is a cosine measure. It is also possible to calculate the Dice or Jaccard coefficient from the following statistics: n_{jk}, n_j, and n_k. When there are large differences in the document frequencies between two words, then $n_{jk}/\min(n_j, n_k)$ may be more useful; this is called the overlapped coefficient. For example, if n_{jk}=10, n_j=1,000, and n_k=10, then the overlap coefficient is 1.0, thus indicating that w_k always co-occurs with w_j. However, the cosine coefficient becomes small (i.e., 0.1) because it is affected by the large document frequency of w_j.

The co-occurrence-based similarity method is advantageous for its simplicity in obtaining the data needed for calculation. For example, it is easy to determine three numbers (i.e., n_{jk}, n_j, and n_k) using an IR system (e.g., a database searching service). This is a primary reason that co-word analyses conducted for bibliometric studies usually employ the co-occurrence-based similarity method.

In the IR field, co-occurrence-based similarity is sometimes used for query expansion. Basically, words (or phrases) with high degrees of similarity to an original query word are automatically added to the query under the assumption that documents including words similar to those in a user query are also relevant to the query. When computing similarity, word co-occurrences are usually counted within a set of particular sentences rather than whole documents (e.g., Jing & Croft, 1994; Xu & Croft, 1996). Further, Mandala, Tokunaga, and Tanaka (1999) adopted a variable-length window size when calculating word co-occurrences.

Unfortunately, previous research has found that query expansions achieved through co-occurrence-based similarity have almost no effect on improving search performance in the IR field (e.g., Peat & Willett, 1991). This is partly because newly added words tend to appear in relevant documents that were already retrieved through the original query words. For example, suppose that w_j appears in an original query; even if w_k is added to the query based on its co-occurrence-based similarity to w_j, it is still difficult for w_k to help detect relevant documents in which original w_j does not appear. This is because high co-occurrence-based similarity entails that the two words will co-occur in many documents.

2.2. Context-Based Distributional Similarity

A basic assumption of distributional similarity is that two words are semantically similar if they share common context words. For example, consider the two following sentences:

(A) Last night, I observed Mercury with my telescope.
(B) During the night, we observed Jupiter by using Martin's telescope.

It is possible to detect a high similarity between "Mercury" and "Jupiter" by focusing on the common context words of "night," "observed," and "telescope."

Indeed, Chen, Fankhauser, Thiel, and Kamps (2005) measured word similarity by counting context words to automatically construct a thesaurus, while Terra and Clarke (2003) examined word similarity as computed through a co-occurrence-based method involving each target word and a common context word. These studies are an example of measuring word similarity based on simply counting context words. Meanwhile, Lin, Sun, Wu, and Xiong (2016) attempted to represent a word vector from a set of its context words, which was then applied to a clustering of tweets. Likewise, Liebeskind et al. (2018) used a vector of context words to automatically construct a Hebrew thesaurus in which cosine and Dice coefficients were employed. The context word vector is useful for computing a word similarity as discussed later. Finally, Pekar and Staab (2003) measured the distance between two nouns based on their collocation with verbs (a similar idea).

A word similarity measure based on context words can easily be estimated as a cosine coefficient between row or column vectors in a word-by-word matrix, as follows:

$$\mathbf{M} = [n_{jk}], j, k = 1, \ldots, M,$$

where M is the total number of different words in a target corpus. However, the row (or column) vectors are generally sparse and high dimensional. This sometimes leads to a computational problem that was solved in a study by Schütze and Pedersen (1997), where word clustering and singular-value decomposition (SVD) were used to reduce the issue of high dimensionality (see below for details); each word was finally represented as a 20-dimensional real-valued "thesaurus vector," which was then used to express documents in an IR algorithm.

The thesaurus vector is a type of word feature vector that allows us to obtain word similarity measures by computing cosine coefficients between vectors, as follows:

$$s_{jk} = \frac{\mathbf{x}_j^T \mathbf{x}_k}{\| \mathbf{x}_j \| \| \mathbf{x}_k \|} \tag{2}$$

where $\mathbf{x}_j$ and $\mathbf{x}_k$ are feature vectors of word w_j and w_k, respectively. Elements of the feature vector are usually real numbers, which provide a distributional representation of each word. This kind of word similarity is often referred to as a distributional similarity.

Because each element of the word feature vector corresponds to a latent topic (or sense) inherent to a given corpus, any mismatches between context words that are caused by surface descriptive variants when detecting common context words is expected to be compensated for through the use of feature vectors. Take examples (A) and (B) above; if "binoculars" is used as a context word in another sentence, then it is not matched with "telescope" in these sentences even though both words refer to optical devices that are often employed in similar situations. When one or more elements in the word feature vector imply such similar device types, an overlap in sentence contexts would be more successfully reflected by a similarity value that is computed by the feature vectors.

An occurrence frequency of word w_j in document d_i is denoted by x_{ij} ($i = 1, \ldots, N$; $j = 1, \ldots, M$), where N indicates the total number of documents in a given corpus. $N \times M$ matrix $\mathbf{X} = [x_{ij}]$ can be decomposed so that $\mathbf{X}^T = \mathbf{AVC}^T$, which is an SVD. A diagonal element of $\mathbf{V}$ corresponds to a latent topic (or sense); it is then possible to interpret the j-th row of $\mathbf{A}$ as a feature vector of word w_j (i.e., a thesaurus vector in Schütze & Pedersen, 1997). If extracting only the L largest diagonal elements in $\mathbf{V}$ ($L < \min(N, M)$), then documents are represented by only "major" latent topics. This is an important procedure in latent semantic analysis (LSA) (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990), which was actually applied to measure word similarities when constructing thesauri from corpora (e.g., Lagutina, Larionov, Petryakov, Lagutina, & Paramonov, 2018; Mohsen, Al-Ayyoub, Hmeidi, & Al-Aiad, 2018). When $\mathbf{V}$ is omitted, the decomposition is then as follows:

$$\mathbf{X}^T \cong \mathbf{AC}^T \tag{3}$$

which is termed a nonnegative matrix factorization (NMF) if $\mathbf{A}$ and $\mathbf{C}$ are nonnegative matrices.

Hofmann (1999) proposed probabilistic LSA (PLSA) that estimates a sequence of probabilities for each word,

$$p(w_j \mid z_1), p(w_j \mid z_2), \ldots, p(w_j \mid z_L),$$

from a corpus in which z_h denotes a latent topic ($j = 1, \ldots, M$; $h = 1, \ldots, L$). The sequence of real numbers can also be treated as a feature vector of each word. It is then possible to employ other models (e.g., the latent Dirichlet allocation [LDA] developed by Blei, Ng, and Jordan, 2003) to obtain a word feature vector consisting of probability $p(w_j \mid z_h)$.

Borrowing terms from Waltz and Pollack (1985) that discussed the human understanding of the world from a cognitive science perspective, each element of a word feature vector corresponds to a "microfeature" of a concept that is represented by the word; its value indicates a level of "activation" of the microfeature. Gallant et al. (1992) were inspired by Waltz and Pollack (1985) in their construction of 300-dimensional vectors to represent words in their own IR system. This may be regarded as an attempt that was relevant to the early stages of the neural network approach, although elements of the vector were determined by simply using data related to word frequencies in a document set.

The distributional representation of words appears in a multilayer model of neural networks when applying the model to textual data. This means that each word in an input text is embedded as a real-valued vector in the network. Word embedding plays a key role in implementing an artificial intelligence system based on a deep learning model when input data are textual. Usually, the distributional representation is computed by applying an algorithm to a large-scale corpus (e.g., Wikipedia) independently from training data that are inherently prepared for the learning process. Word2Vec is a particularly well-known algorithm in this context.

An assumption of Word2Vec is that words appearing with a similar context are similar. As such, any feature vectors estimated through Word2Vec can be used to compute context-based distributional word similarities. For example, the vector can be directly used as $\mathbf{x}_j$ in Equation 2. Both Shunmugam and Archana (2016) and Poostchi and Piccardi (2018) incorporated word feature vectors obtained through Word2Vec into a k-means algorithm for word clustering.

Other algorithms for word embedding have been also developed. For instance, Pennington, Socher, and Manning (2014) proposed an algorithm based on global vectors (GloVe) constructed from global corpus statistics, whereas a shallower window is used to count the co-occurrences of a target word and its context word in Word2Vec. Similarly, Zhao, Liu, Li and Du (2016) explored a word embedding algorithm designed to consider context words within a full document range rather than a given sentence. There are also other word-embedding algorithms that enable users to compute distributional similarities between words.

2.3. Similarity Based on a Vector Space Model

According to the VSM developed by G. Salton's research group, a word is often represented as a vector of which element is a weight of the word in each document. For example, element v_{ij} is defined as a tf-idf weight, as follows:

$$v_{ij} = x_{ij} \log \frac{N}{n_j} \tag{4}$$

which constructs N-dimensional feature vector $\mathbf{x}_j = [v_{1j}, v_{2j}, \ldots, v_{Nj}]^T$ for word $w_j (j = 1, \ldots, M)$. The cosine coefficient between vectors (i.e., Equation 2) has often been used to create an M

× M similarity matrix to achieve query expansion in the IR field. The resulting matrix is termed a similarity thesaurus (Qiu & Frei, 1993). If the similarity matrix is denoted by **W**, then M-dimensional query vector $\mathbf{q}$ is modified so that $\tilde{\mathbf{q}} = \mathbf{Wq}$ (note that matrix **W** can be also generated through a co-occurrence similarity method).

Zazo et al. (2005) proved that query expansion achieved through the similarity thesaurus had a positive effect when search queries were short, while Mohsen et al. (2018) used a similarity thesaurus to expand queries in the Arabic language. Similarity thesauri have also been used to improve machine learning performance. For example, Xu and Yu (2010) used a similarity thesaurus to detect spam e-mails through a neural network model, while Li, Yang, and Park (2012) used one to enhance the effectiveness of text categorization.

Studies have also explored different similarity measures between word vectors achieved through documents in which the word appears. For example, Jo (2016) represented a word using a set of documents in which it was included. Here, the similarity between two words was measured based on the similarity between the two document sets.

3. EXPERIMENTAL PROCEDURE FOR COMPARING WORD SIMILARITY MEASURES

This study conducted an experiment to empirically identify the characteristics of the three types of word similarity measures reviewed in Section 2, including the a) co-occurrence-based, b) context-based distributional, and c) VSM-based vector type used to construct similarity thesauri. More specifically, the real values of these similarity measures were computed from a set of documents extracted from the RCV1 for direct comparison.

3.1. Data

Word similarities were computed using a total of 6,374 records to which a single topical code was assigned in a set of news articles published between August 20-31, 1996. This was used as a test set to explore the document clustering algorithm in an experiment by Kishida (2011). However, this study placed no special implications on the restringing of articles to which a single topical code was assigned. Rather, this was only done because the articles had been intensively checked by an author prior to this experiment.

Nouns, verbs, adjectives, adverbs, cardinal numbers, and foreign words were extracted from the headlines and main texts of each news article using version 3.9.2 of the Stanford POS tagger (Toutanova, Klein, Manning, & Singer, 2003).

3.2. Comparative Analysis Methods

Word similarity measures were empirically compared via two metrics of proximity or difference, as follows:

(A) Pearson correlation coefficient values, which were calculated between two matrices of similarity measures

(B) normalized mutual information (nMI) scores, which were calculated between two sets of document groups that were generated from individual similarity measures through a clustering algorithm

The Pearson correlation coefficient was standardly computed after vectorizing target matrices, meaning that the coefficient was directly calculated from corresponding pairs of $M(M$-1)/2 elements of two matrices. While this was a direct comparison, an indirect comparison was achieved by using the nMI scores between clustering results. An examination of clustering results is useful for applications in which word clusters have important functions. Although several ways of normalization can be used to define nMI (Kishida, 2014), this experiment normalized MI scores according to maximum entropy values.

Each word was stemmed through the Porter algorithm. A resulting list of word stems was then created in descending order of document frequency (i.e., n_j). Among those appearing in 100 or more news articles, the authors then intentionally selected 50 words that unambiguously represented a concept belonging to only one of the five following topics: (a) the economy, (b) politics, (c) crime, (d) war, and (e) sports. These categories were determined after carefully examining the top-ranked word stem list; the authors did not find any topic to which 10 or more words belong other than the five topics in the dataset. Table 1 shows the 50 total word stems across all five topics.

It was expected that a comparative analysis would be easier to conduct when only considering unambiguous words even

Table 1. All 50 word stems selected for comparison

Economy	Politics	Crime	War	Sports
bond	communist	arrest	armi	champion
cash	congress	crimin	attack	cricket
debt	democrat	jail	bomb	game
dollar	diplomat	kidnap	fight	leagu
export	govern	law	guerrilla	player
import	minist	legal	militari	soccer
invest	parliament	murder	rebel	sport
market	polit	prison	soldier	team
monei	politician	prosecutor	troop	tenni
trade	republ	victim	war	tournament

though it was also important to examine ambiguous words (i.e., those with multiple meanings). Although some words were equally related to two topic categories (e.g., “bomb” may be related to both “crime” and “war”), the authors considered that the 50 stems were adequate for this experiment. Note that the five groups shown in Table 1 could be used as a ground truth, thus allowing an external evaluation of all clustering results. This is another benefit of specifically selecting a set of unambiguous words as a target group.

3.3. Computing Word Similarity Measures

Table 2 shows the word similarity measures examined in this study’s experiment. The co-occurrence-based similarity was computed through Equation 1 in two cases where the numbers of sentences and documents were used for n_{jk}, n_j, and n_k, respectively. On the other hand, the context-based distributional similarity was calculated through the NMF, LDA, and Word2Vec frameworks, which generated word feature vectors $\mathbf{x}_j$ ($j = 1, \ldots, M$) in Equation 2, respectively. The number of dimensions of the feature vectors was commonly set to 100, thus corresponding to the number of latent topics in both NMF and LDA (i.e., $L = 100$). Word feature vectors were also constructed by using Equation 4, from which VSM-based word similarity was calculated by Equation 2. Note that the NMF and LDA algorithms were executed after removing words (stems) that only appeared in one article. They were also deleted from word feature vectors in VSM to maintain the same condition.

This experiment only used a cosine measure in Equation 1 or 2, although it was also possible to calculate the Dice, Jaccard, or overlapped coefficients. Further, information-theoretic measures (e.g., the point-wise mutual information, Kullback-Leibler divergence, and Jensen-Shannon divergence) are often employed when measuring word similarities used for natural language processing (NLP) (Dagan et al., 1999). However, these were outside the scope of this study, which also excluded nonsymmetric measures explored in the NLP field (Kotlerman, Dagan, Szpektor, & Zhitomirsky-Geffet, 2010). As such, this experiment solely focused on the cosine measure widely applied throughout the IR and bibliometrics (scientometrics) fields.

After computing s_{jk} in Equation 1 or 2 for the 50 stems shown in Table 1 (j, $k = 1, \ldots, 50$) according to the individual definitions of six similarity measures in Table 2, a Pearson correlation coefficient was calculated for each pair. Because $M = 50$, the sample size in calculation of the coefficient was 1,225 (= 50 × 49 ÷ 2). Next, a group-average agglomerative hierarchical clustering (AHC) was executed for each set of similarity values s_{jk} after converting it to a distance metric so that $1.0 - s_{jk}$. This experiment only used the group-average method because it clearly outperformed a complete linkage method during a preliminary analysis. Classical multidimensional scaling (MDS) was also partly used to observe visual proximities between the words derived from each word similarity measure.

3.4. Experimental System

For the computational process, the Word2Vec algorithm was executed for the test set using an Apache Spark module (Word2Vec class). The hclust and cmdscale fuctions of R packages (version 3.6.1) were then applied for the AHC and MDS, respectively. Other computer processing modules were constructed using the Java language. Probability $p(w_j \mid z_k)$ in the LDA model was estimated via Gibbs sampling (Griffiths & Steyvers, 2004). More specifically, in the r-th iteration of

Table 2. Word similarity measures

Word similarity s_{jk}	Acronym
(A) Co-occurrence-based similarity	
(i) Using the number of sentences in Equation 1	CoocS
(ii) Using the number of documents in Equation 1	CoocD
(B) Context-based distributional similarity	
(i) Using a row of **A** in NMF (see Equation 3) as feature vector $\mathbf{x}_j$ in Equation 2	NMF
(ii) Using sequence $p(w_j \mid z_1), \ldots, p(w_j \mid z_L)$ obtained by LDA as feature vector $\mathbf{x}_j$ in Equation 2	LDA
(iii) Using word embedding through the Word2Vec algorithm as feature vector $\mathbf{x}_j$ in Equation 2	W2V
(C) VSM-based similarity for similarity thesauri	
(i) Using N dimensional vector of which element is a tf-idf weight in Equation 4 as feature vector $\mathbf{x}_j$ in Equation 2	VSM

L refers to the number of columns in **A**, which corresponds to the number of latent topics in LDA, while N denotes the number of documents.
NMF, nonnegative matrix factorization; LDA, latent Dirichlet allocation; VSM, vector space model.

Table 3. Predetermined parameters

Word similarity		Predetermined parameters
Co-occurrence-based	CoocS, CoocD	None
Distributional	NMF	Iterations: 100
	LDA	Hyperparameters: $\alpha = 0.1$, $\beta = 0.01$ Iterations: 2,100 (burn-in period: 100)
	W2V	Max iterations: 2,000, Window-size: 5
VSM-based	VSM	$N = 6{,}374$

CoocS, co-occurrence-based similarity according to number of sentences; CoocD, co-occurrence-based similarity according to number of documents; NMF, nonnegative matrix factorization; LDA, latent Dirichlet allocation; W2V, Word2Vec; VSM, vector space model.

the sampling, $p_r(w_j \mid z_k)$ was computed as a percentage of w_j in tokens to which the k-th latent topic was allocated. After R iterations, $p(w_j \mid z_k)$ was estimated as an average so that $p(w_j \mid z_k) = R^{-1} \sum_{r=1}^{R} p_r\,(w_j \mid z_k)$ except for iterations during the burn-in period. Finally, NMF was obtained via an algorithm developed by Lee and Seung (1999) under the condition that a row of **A** was normalized by its norm. The predetermined parameters for each process are shown in Table 3.

4. EXPERIMENTAL RESULTS OF COMPARING WORD SIMILARITY MEASURES

A total of 26,594 stems of nouns, verbs, adjectives, adverbs, cardinal numbers, and foreign words were obtained from 6,374 news articles after removing those that only appeared in one article (N = 6,374; M = 26,594). Thus, a total of 1,237,831 tokens were included in our test set (i.e., collection length), meaning that the average document length was 194.2. This set of sentences and documents was then used to calculate word similarity measures (Table 2) for the 50 selected words (stems) (Table 1).

4.1. Comparison Based on a Pearson Correlation Coefficient

Table 4 shows the Pearson correlation coefficient values between the six word-similarity measures computed from each set of 1,225 similarity pairs between all 50 stems, as described in Section 3.3. The closest similarity measures were the co-occurrence-based similarity according to number of documents (CoocD) and VSM-based similarity (VSM), of which the value was 0.939. On the other hand, the value was lowest (0.550) between the co-occurrence-based similarity according to number of sentences (CoocS) and NMF-based similarity (NMF).

Table 4. Pearson correlation coefficient between word similarity measures

	CoocS	CoocD	NMF	LDA	W2V
CoocD	0.849				
NMF	0.550	0.681			
LDA	0.689	0.766	0.607		
W2V	0.592	0.645	0.562	0.664	
VSM	0.909	0.939	0.645	0.795	0.644

CoocS, co-occurrence-based similarity according to number of sentences; CoocD, co-occurrence-based similarity according to number of documents; NMF, nonnegative matrix factorization; LDA, latent Dirichlet allocation; W2V, Word2Vec; VSM, vector space model.

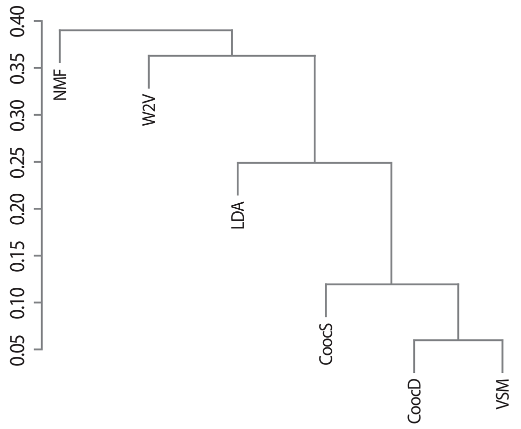

Fig. 1. Agglomerative hierarchical clustering results according to a Pearson correlation coefficient. NMF, nonnegative matrix factorization; W2V, Word2Vec; LDA, latent Dirichlet allocation; CoocS, co-occurrence-based similarity according to number of sentences; CoocD, co-occurrence-based similarity according to number of documents; VSM, vector space model.

Fig. 1 shows the result of group-average AHC executed for the correlation matrix shown in Table 4 after each correlation value was simply converted to a distance metric (i.e., = 1.0–correlation). The relatedness among similarity measures is clearly demonstrated through the dendrogram in Fig. 1, which indicates that the co-occurrence-based similarity (CoocS and CoocD) and VSM-based similarity measure formed a group. These are traditional measures that have been used for a very long time in the IR field. On the other hand, the LDA-based similarity measure was relatively near the group, while the Word2Vec- and NMF-based similarity measures differed from those.

4.2. Comparing Clustering Results

As an example, two MDS plots for the data from the VSM-based and Word2Vec-based similarity measures are shown in

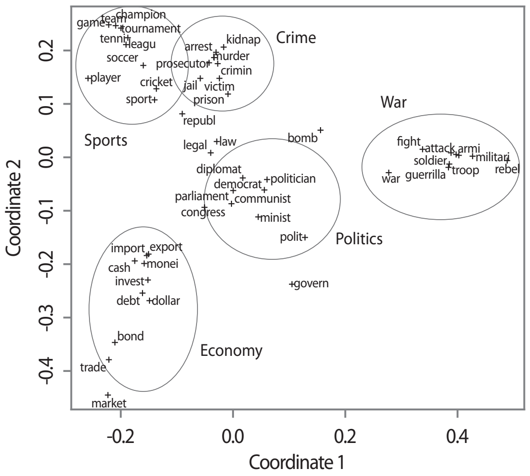

Fig. 2. Multidimensional scaling plot (vector space model-based similarity measure).

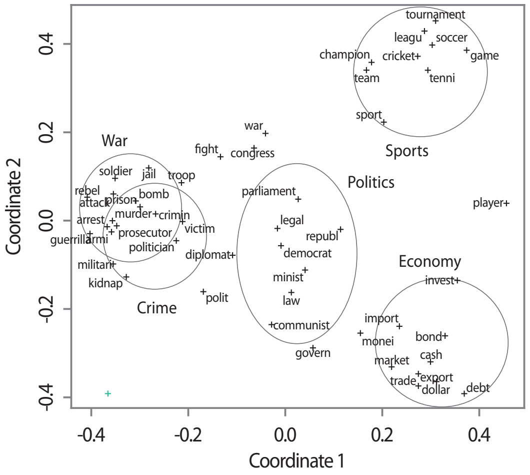

Fig. 3. Multidimensional scaling plot (Word2Vec-based similarity measure).

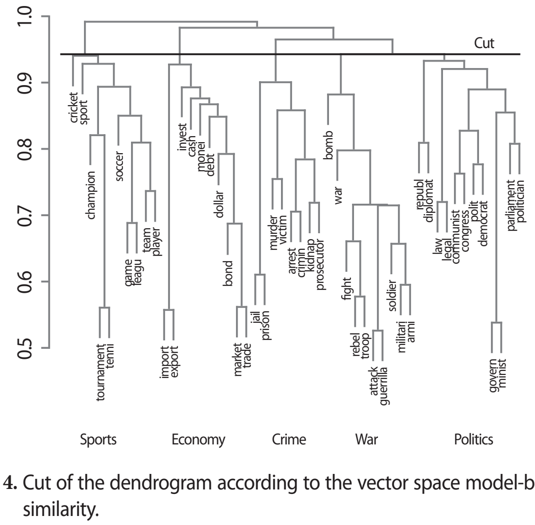

Fig. 4. Cut of the dendrogram according to the vector space model-based similarity.

Figs. 2 and 3, respectively. Although the crime and war word groups overlapped in the MDS plot according to the Word2Vec-based similarity measure (Fig. 3), there appeared to be no large differences between the two maps.

A typical AHC clustering result is shown in Fig. 4 through a dendrogram that was drawn using data from the VSM-based similarity measure, in which five clusters were generated through a cut operation (i.e., the cutree function of the R package). Clustering was successful because the words in each subtree (except for the words "law" and "legal") corresponded to one of the groups shown in Table 1.

Table 5 shows the nMI scores among the word-group sets (i.e., cluster sets) generated by cutting the dendrograms obtained from data of individual similarity measures when the number of clusters was set to be five (i.e., $H = 5$ in which H means the number of word clusters). As shown, a set of the five groups in Table 1 were also included as an "Answer" that could be employed as a ground truth provided by human annotators. According to this ground truth, the most successful clustering result was obtained from the VSM-based similarity measure, followed by the co-occurrence and LDA-based measures (CoocS, CoocD, and LDA). However, the NMF-based measure provided the lowest nMI score with the "Answer" in this experiment.

The exact number of clusters is unknown in many situations involving document clustering. This experiment therefore attempted to increase the numbers of clusters from five to 10 (i.e., $H = 5,6,7,8,9,10$) although 50 words were intentionally selected from five total topics. Fig. 5 shows the nMI scores between cluster sets and the "Answer" according to the number of clusters. Here, it is evident that the VSM- and LDA-based similarity measures produced better overall clustering results. Note that the number of clusters in the "Answer" was always fixed to five when calculating the nMI scores shown in Fig. 5.

Table 5. Normalized mutual information scores between word-group sets according to word similarity measures ($H = 5$)

	CoocS	CoocD	NMF	LDA	W2V	VSM
CoocD	0.930					
NMF	0.630	0.644				
LDA	0.930	1.000	0.644			
W2V	0.753	0.691	0.583	0.691		
VSM	0.764	0.764	0.597	0.764	0.699	
Answer	0.787	0.787	0.590	0.787	0.715	0.957

H means the number of word clusters.
CoocS, co-occurrence-based similarity according to number of sentences; CoocD, co-occurrence-based similarity according to number of documents; NMF, nonnegative matrix factorization; LDA, latent Dirichlet allocation; W2V, Word2Vec; VSM, vector space model.

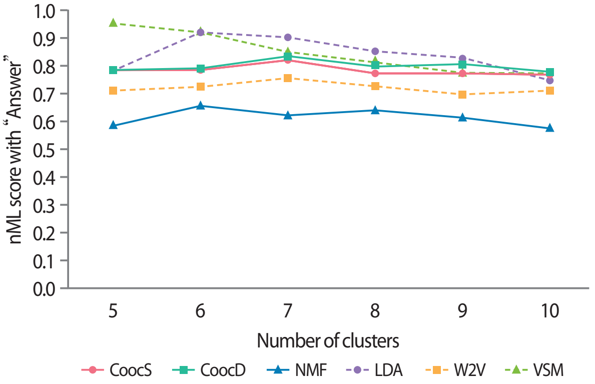

Fig. 5. Normalized mutual information (nMI) scores with "Answer" cluster sets (*H* = 5,6,7,8,9,10). *H* means the number of word clusters. CoocS, co-occurrence-based similarity according to number of sentences; CoocD, co-occurrence-based similarity according to number of documents; NMF, nonnegative matrix factorization; LDA, latent Dirichlet allocation; W2V, Word2Vec; VSM, vector space model.

Table 6. Average normalized mutual information scores between word group sets according to word similarity measures (*H* = 5 to 10)

	CoocS	CoocD	NMF	LDA	W2V	VSM
CoocD	0.853					
NMF	0.664	0.676				
LDA	0.833	0.849	0.650			
W2V	0.713	0.683	0.615	0.743		
VSM	0.788	0.821	0.663	0.895	0.714	
Answer	0.786	0.802	0.620	0.843	0.726	0.851

H means the number of word clusters.
CoocS, co-occurrence-based similarity according to number of sentences; CoocD, co-occurrence-based similarity according to number of documents; NMF, nonnegative matrix factorization; LDA, latent Dirichlet allocation; W2V, Word2Vec; VSM, vector space model.

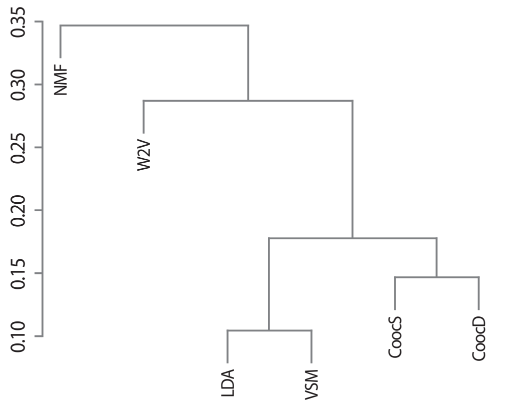

Fig. 6. Results of agglomerative hierarchical clustering according to average normalized mutual information scores. NMF, negative matrix factorization; W2V, Word2Vec; LDA, latent Dirichlet allocation; VSM, vector space model; CoocS, co-occurrence-based similarity according to number of sentences; CoocD, co-occurrence-based similarity according to number of documents.

The averages of nMI scores from $H = 5$ to 10 are shown in Table 6. That is, while the nMI scores in Table 5 only show cases in which $H = 5$, Table 6 provides the results of averaging the six nMI scores for each pair of cluster sets. Fig. 6 also shows a dendrogram that was generated using a set of the average nMI scores as a similarity matrix (except for the "Answer"). The dendrogram in Fig. 6 is similar to that shown in Fig. 1, which was drawn according to Pearson correlation coefficient values. The co-occurrence-, LDA-, and VSM-based similarity measures form a cluster that is remotely located from the Word2Vec- and NMF-based measures. Specifically, a pair of LDA- and VSM-based similarity measures was strongly related due to the generation of similar cluster sets. Likewise, the two co-occurrence-based measures of CoocS and CoocD were closely located in the dendrogram. This was predictable because the only difference between them was found in the ranges of textual data when counting co-occurrence frequencies.

5. DISCUSSION

As shown in Figs. 1 and 6, the co-occurrence- (CoocS and CoocD), LDA-, and VSM-based similarity measures represented relatively similar relationships between words, while the Word2Vec and NMF algorithms provided different similarities. Because the multiplication of tf-idf values (v_{ij}) of a word between two documents becomes zero if the word does not appear in one document, it is easy to conjecture a resemblance between the VSM- and co-occurrence-based similarity measures in view of the number of documents (CoocD).

If all tf values within a given corpus are commonly 1 (i.e., $x_{ij} = 1$), then the inner product of the VSM-based similarity measure in the numerator of Equation 2 is as follows:

$$\mathbf{x}_j^T\mathbf{x}_k = \sum_{i=1}^{N} x_{ij}\log\frac{N}{n_j}x_{ik}\log\frac{N}{n_k} = n_{jk}\times\log\frac{N}{n_j}\times\log\frac{N}{n_k} = n_{jk}\,r_j\,r_k \quad (5)$$

where $r_j = \log\frac{N}{n_j}$ and $r_k = \log\frac{N}{n_k}$. Because $x_{ij} = 1$, the cosine measure is computed as follows:

$$s_{jk} = n_{jk}\times\frac{r_j}{\sqrt{n_j r_j^2}}\times\frac{r_k}{\sqrt{n_k r_k^2}} = \frac{n_{jk}}{\sqrt{n_j}\sqrt{n_k}},$$

which is equal to the co-occurrence-based similarity measure. If x_{1j} changes to $x_{1j} + 1$ in document d_1, then the difference of the inner product in Equation 5 is as follows:

$$\Delta(\mathbf{x}_j^T\mathbf{x}_k) = \log\frac{N}{n_j} \times x_{1k}\log\frac{N}{n_k} \qquad (6)$$

Equation 6 suggests that the inner product increases depending on the tf of the other word and idf values of two words. In fact, the amount of change in s_{jk} is more complicated because a value of the denominator of Equation 2 (i.e., $\|\mathbf{x}_j\|\|\mathbf{x}_k\|$) also varies with $x_{1j} \rightarrow x_{1j} + 1$.

Further, the definitions of CoocS and CoocD become equivalent if all documents each consist of single sentences (i.e., short texts). In a corpus of such short documents, the tf of each word would be near 1, for which there would be only a small difference between the VSM- and co-occurrence-based similarity measures. Although the news articles with headlines and main texts used in this study were not short (average document length was 194.2), the values of the VSM- and co-occurrence-based similarity measures were relatively similar (Fig. 1).

Regarding the clustering results, the VSM- and LDA-based similarity measures generated similar cluster sets (Fig. 6); these were also near the answer set (Table 6). The true answer of clustering words is highly dependent on the target corpus, meaning that the grouping of 50 words shown in Table 1 is not always true when used as the "Answer." For example, there may be a corpus in which "cash" appears in only documents categorized as "sports." Clustering results also change according to the clustering algorithm. The better performance shown from the VSM- and LDA-based similarity measures must be interpreted in consideration of these limitations.

News articles tend to contain a definite target topic (e.g., the economy or sports). This means that documents (not words) were clearly partitioned into topic groups. The LDA model included a document-based probability ($p(z_k \mid d_i)$), while VSM-based similarity was also measured from document vectors. Such document-linked architecture may have contributed to the better performance found with the LDA- and VSM-based similarity measures in the experiment using the news article set. Further, the Word2Vec algorithm only checked for co-occurrences within a small window in each text (the window size was set to five in this experiment). As such, the benefits of using topically cleared news articles could not be incorporated, thus possibly resulting in the lower performance found in this experiment.

6. CONCLUSION

This paper reported on the results of an experiment designed to examine word similarity measures using a portion of the RCV1. We thereby found similar values between a co-occurrence-based similarity measure and one based on tf-idf weights (i.e., a VSM-based measure). We also compared cluster sets generated by the average-group AHC algorithm from individual similarity matrices, thereby finding better clustering results through the VSM- and LDA-based similarity measures. On the other hand, the Word2Vec- and NMF-based similarity measures differed from the other tested measures. From a practical viewpoint, this experiment suggested that word similarity measures computed by LDA and VSM are expected to enhance effectiveness of query expansion or related applications because it is considered that they can identify 'true' similar words more correctly. While VSM is known to generate such effective similarities, it is interesting that LDA also works well.

As discussed above, this study had some limitations. For one, we only used a set of news articles to compute similarity measures. Two, only an AHC algorithm was applied to the similarity matrices thus obtained. Three, symmetric similarity measures other than the cosine and nonsymmetric similarity measures were outside the scope of this study's experiment. As such, future studies should conduct experiments from both the theoretical and empirical viewpoints to gain a deeper understanding of the tested word similarity measures.

REFERENCES

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3, 993-1022.

Chen, L., Fankhauser, P., Thiel, U., & Kamps, T. (2005, October 31-November 5). Statistical relationship determination in automatic thesaurus construction. In O. Herzog, H.-J. Schek, N. Fuhr, A. Chowdhury, & W. Teiken (Eds.), *Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM '05)* (pp. 267-268). Association for Computing Machinery.

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

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6), 391-407.

Gallant, S., Hecht-Nielson, R., Caid, W., Qing, K., Carleton, J., & Sudbeck, D. (1992, November 4-6). HNC's MatchPlus System. In D. K. Harman (Ed.), *The First Text REtrieval*

Conference (TREC-1) (pp. 107-111). National Institute of Standards and Technology.

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. Proceedings of the National Academy of Sciences of the United States of America, 101(suppl 1), 5228-5235.

Hofmann, T. (1999, August 15-19). Probabilistic latent semantic indexing. In F. Gey, M. A. Hearst, & R. Tong (Eds.), *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '99)* (pp. 50-57). Association for Computing Machinery.

Jing, Y., & Croft, W. B. (1994, October 11-13). An association thesaurus for information retrieval. In J-L. F. Brentano & F. Seitz (Eds.), *RIAO '94: Intelligent Multimedia Information Retrieval Systems and Management – Vol. 1* (pp. 146-160). Le Centre de Hautes Etudes Internationales d'Informatique Documentaire.

Jo, T. (2016, July 25-28). String vector based AHC as approach to word clustering. In R. Stahlbock & G. M. Weiss (Eds.), *Proceedings of the International Conference on Data Mining DMIN'16* (pp. 133-138). Lancaster Centre for Forecasting.

Khasseh, A. A., Soheili F., Moghaddam, H. S., & Chelak, A. M. (2017). Intellectual structure of knowledge in iMetrics: A co-word analysis. *Information Processing & Management*, 53(3), 705-720.

Kishida, K. (2011). Double-pass clustering technique for multilingual document collections. *Journal of Information Science*, 37(3), 304-321.

Kishida, K. (2014). Empirical comparison of external evaluation measures for document clustering by using synthetic data. *IPSJ SIG Technical Report*, 2014-IFAT-113, 1-7.

Kotlerman, L., Dagan, I., Szpektor, I., & Zhitomirsky-Geffet, M. (2010). Directional distributional similarity for lexical inference. *Natural Language Engineering*,16(4), 359-389.

Lagutina, K., Larionov, V., Petryakov, V., Lagutina, N., & Paramonov, I. (2018, November 13-16). Sentiment classification of Russian texts using automatically generated thesaurus. In S. Balandin, T. S. Cinotti, F. Viola, & T. Tyutina (Eds.), *Proceedings of the 23rd Conference of Open Innovations Association FRUCT* (pp. 217-222). FRUCT Oy.

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401, 788-791.

Li, C. H., Yang, J. C., & Park, S. C. (2012). Text categorization algorithms using semantic approaches, corpus-based thesaurus and WordNet. *Expert Systems with Applications*, 39(1), 765-772.

Liebeskind, C., Dagan, I., & Schler, J. (2018, May 7-12). Automatic thesaurus construction for modern Hebrew. In N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, & T. Tokunaga (Eds.), *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)* (pp. 1446-1451). European Language Resources Association.

Lin, H., Sun, B., Wu, J. & Xiong, H. (2016, June 24-26). Topic detection from short text: A term-based consensus clustering method. In B. Yang (Ed.), *2016 13th International Conference on Service Systems and Service Management (ICSSSM 2016)* (pp. 1-6). IEEE.

Mandala, R., Tokunaga, T., & Tanaka, H. (1999, August 15-19). Combining multiple evidence from different types of thesaurus for query expansion. In F. Gey, M. A. Hearst, & R. Tong (Eds.), *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 191-197). Association for Computing Machinery.

Mikolov, T., Yih, W. T., & Zweig, G. (2013, June 9-14). Linguistic regularities in continuous space word representations. In L. Vanderwende, H. Daumé III, & K. Kirchhoff (Eds.), *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 746-751). Association for Computational Linguistics.

Mohsen, G., Al-Ayyoub, M., Hmeidi, I., & Al-Aiad, A. (2018, April 3-5). On the automatic construction of an Arabic thesaurus. *2018 9th International Conference on Information and Communication Systems (ICICS)* (pp. 243-247). IEEE.

Peat, H. J., & Willett, P. (1991). The limitations of term co-occurrence data for query expansion in document retrieval systems. *Journal of the American Society for Information Science*, 42(5), 378-383.

Pekar, V., & Staab, S. (2003, April 12-17). Word classification based on combined measures of distributional and semantic similarity. In A. Copestake & J. Hajič (Eds.), *EACL '03: Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics) – Vol. 2* (pp. 147-150). Association for Computational Linguistics.

Pennington, J., Socher, R., & Manning, C. D. (2014, October 25-29). GloVe: Global vectors for word representation. In A. Moschitti, B. Pang, & W. Daelemans (Eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532-1543). Association for Computational Linguistics.

Poostchi, H., & Piccardi, M. (2018, December 10-12). Cluster labeling by word embeddings and WordNet's hypernymy. In S. M. Kim & X. Zhang (Eds.), *Proceedings of the Australasian Language Technology Association Workshop 2018* (pp. 66-70). Association for Computational Linguistics.

Qiu, Y., & Frei, H. -P. (1993, June 27-July 1). Concept based query expansion. In R. Korfhage, E. M. Rasmussen, & P. Willett (Eds.), *SIGIR '93: Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 160-169). Association for Computing Machinery.

Ravikumar, S., Agrahari, A., & Singh, S. N. (2015). Mapping the intellectual structure of scientometrics: A co-word analysis of the journal Scientometrics (2005-2010). *Scientometrics*, 102(1), 929-955.

Schütze, H., & Pedersen, J. O. (1997). A cooccurrence-based thesaurus and two applications to information retrieval. *Information Processing & Management*, 33(3), 307-318.

Shunmugam, D. A., & Archana, P. (2016). An empirical investigation of word clustering techniques for natural language understanding. *International Journal of Engineering Science and Computing*, 6(10), 2637-2646.

Terra, E. L., & Clarke, C. L. A. (2003, May 27-June 1). Frequency estimates for statistical word similarity measures. In M. Hearst & M. Ostendorf (Eds.), *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology – Vol. 1* (pp. 165-172). Association for Computational Linguistics.

Toutanova, K., Klein, D., Manning, C. D., & Singer, Y. (2003, May 27-June 1). Feature-rich part-of-speech tagging with a cyclic dependency network. In M. Hearst & M. Ostendorf (Eds.), *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology – Vol. 1* (pp. 173-180). Association for Computational Linguistics.

Waltz, D. L., & Pollack, J. B. (1985). Massively parallel parsing: A strongly interactive model of natural language interpretation. *Cognitive Science*, 9(1), 51-74.

Xu, H., & Yu, B. (2010). Automatic thesaurus construction for spam filtering using revised back propagation neural network. *Expert Systems with Applications*, 37(1),18-23.

Xu, J., & Croft, W. B. (1996, August 18-22). Query expansion using local and global document analysis. In H.-P. Frei, D. Harman, P. Schäuble, & R. Wilkinson (Eds.), *SIGIR '96: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 4-11). Association for Computing Machinery.

Zazo, A. F., Figuerola, C. G., Berrocal, J. L. A., & Rodríguez, E. (2005). Reformulation of queries using similarity thesauri. *Information Processing & Management*, 41(5), 1163-1173.

Zhao, Z., Liu, T., Li, B., & Du, X. (2016, August 29-September 2). Cluster-driven model for improved word and text embedding. In G. A. Kaminka, M. Fox, P. Bouquet, E. Hüllermeier, V. Dignum, & (Eds.), *ECAI'16: Proceedings of the Twenty-second European Conference on Artificial Intelligence* (pp. 99-106). IOS Press.