Journal of Artificial Intelligence Research 72 (2021) 1281-1305 Submitted 09/2021; published 12/2021

A Word Selection Method for Producing Interpretable Distributional Semantic Word Vectors

Atefe Pakzad A_PAKZAD@COMP.IUST.AC.IR
School of Computer Engineering, Iran University of Science and Technology
Tehran, Iran

Morteza Analoui ANALOUI@ IUST.AC.IR
School of Computer Engineering, Iran University of Science and Technology
Tehran, Iran

Abstract

Distributional semantic models represent the meaning of words as vectors. We introduce a selection method to learn a vector space that each of its dimensions is a natural word. The selection method starts from the most frequent words and selects a subset, which has the best performance. The method produces a vector space that each of its dimensions is a word. This is the main advantage of the method compared to fusion methods such as NMF, and neural embedding models. We apply the method to the ukWaC corpus and train a vector space of N=1500 basis words. We report tests results on word similarity tasks for MEN, RG-65, SimLex-999, and WordSim353 gold datasets. Also, results show that reducing the number of basis vectors from 5000 to 1500 reduces accuracy by about 1.5-2%. So, we achieve good interpretability without a large penalty. Interpretability evaluation results indicate that the word vectors obtained by the proposed method using N=1500 are more interpretable than word embedding models, and the baseline method. We report the top 15 words of 1500 selected basis words in this paper.

1. Introduction

Distributional semantics (DS), also known as vector space semantics, is a model that presumes a key role for the statistical distribution of linguistic items in determining their semantic behavior (Lenci, 2018). In the vector space models of word meaning, the distribution of the word's contexts for deriving an appropriate meaning representation is too important (Clark, 2015). The distributional models of meaning build co-occurrence vectors for every word in a corpus, based on its context following Firth's intuition that "you should know a word by the company it keeps" (Kartsaklis, 2014).

At first, we must determine which words will be 'target' words and which words will form 'context' words. The target words are co-occurred with some context words in the corpus. Target and context words are typically selected based on the word's frequency. Second, for constructing "word space", we count the occurrences of our target words within the context words in a corpus. Counts of co-occurrences of target words with context words construct a co-occurrence matrix where the rows are the target words, the columns are the context words, and each cell represents the number of times each target word occurred within the context word (Heunen et al., 2013).

As a small artificial example, given a corpus including sentences like these (Heunen et al., 2013):

Bank erosion and stream widening may occur with strong water flow.

One way of raising this finance is to go a bank.

etc.

We might construct a matrix like:

	Context words				
Target	***River***	***Stream***	***Money***	***Raise***	***Finance***
Bank	10	15	25	20	13
Water	28	25	2	15	0
Cheque	0	0	30	20	25

Table 1: Co-occurrence Matrix (Heunen et al., 2013)

The resulting rows of the co-occurrence matrix can be considered as vectors in a multidimensional space. Words that have similar meanings have vectors with similar directions (Heunen et al., 2013).

1.1. Distributional Semantic Vectors

The distribution of word meaning is a vector in a vector space that the context determines the basis vectors. The vector spaces of distributional semantic models (DSMs) have orthogonal bases. The inner product of each basis vector with others is zero. A semantic vector of a word can be represented as the weighted superposition of the basis vectors (Grefenstette, 2013):

$$\overrightarrow{\text{word}} = \sum_j c_j \overrightarrow{n_j} \quad (1)$$

Where a set of orthogonal unit vectors $\{\overrightarrow{n_j}\}_j$ is the basis of vector space that the word meaning lives in. Parameter c_j is a weight for basis vector $\overrightarrow{n_j}$. We represent these basis vectors with the context words (Grefenstette, 2013). The most basic form of the c_j is co-occurrence counts, but a function on counts is often used to reduce unavoidable frequency bias (Kartsaklis, 2014).

1.2. Positive Pointwise Mutual Information (PPMI)

A measure of association determines how much two words co-occur. Pointwise mutual information (PMI) is a measure that represents how often event x and event y co-occur in contrast to when they were independent (Jurafsky & Martin, 2014):

$$I(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)} \quad (2)$$

We can define the pointwise mutual information association for a target word w and a context word c as

$$PMI(w,c) = \log_2 \frac{P(w,c)}{P(w)P(c)} \quad (3)$$

$P(w,c)$ determines how often two words co-occur. $P(w)P(c)$ informs us that how often we expect the two words to co-occur when they each occur independently. Consequently, this ratio tells us how much more than our expectation the target word w and the context word c co-occur. The range of

PMI values is negative to positive infinity. Negative PMI values indicate that two words are co-occurring less than our expectation by chance. It is more common to use Positive PMI (PPMI) because negative PMI values just are reliable when the corpus is huge. PPMI replaces all negative PMI values with zero (Jurafsky & Martin, 2014).

$$\mathrm{PPMI}(w,c) = \max\left(\log_2 \frac{P(w,c)}{P(w)P(c)}, 0\right) \qquad (4)$$

In this paper, for the first time, we introduce a framework to get basis vectors as N final basis words. First, our proposed method begins with a set of initial basis words, which are the most frequent in the corpus. We start from a 5K initial set and refine it to a final N dimension vector space relying on the concept of distance matrices. Our method produces a vector space that each of its dimensions is a natural word. This is the main difference of the selection method compared to the so-called fusion methods such as NMF and neural embedding models. Basis vectors in the fusion methods and neural embedding have no direct interpretable meaning. The resulting word vectors quality is usually evaluated on word similarity tasks for standard test sets. These test sets include word pairs and corresponding gold standard scores. These scores show the similarities between the words that are found out by human judges via an annotation task (Bruni et al., 2012). We compare predicted word similarity with the gold standard to evaluate the model. We use Spearman's correlation test to evaluate the framework. We also evaluate the qualitative and quantitative interpretability of the proposed method compared to word embedding models and the Baseline method.

We present the related work in the next Section, and we describe the original contribution in Section 3. In Section 4, we describe the importance of dimensional reduction in semantic vectors. Sections 5, 6, and 7 are three main sections. Section 5 fully introduces the proposed framework. We describe the experimental settings in detail in Section 6. A discussion of the results is given in Section 7. The evaluations include word similarity task, qualitative interpretability, and quantitative interpretability are discussed in Subsections 7.1, 7.2, and 7.3, respectively. The conclusion is presented in Section 8.

2. Related Work

Distributional semantic models indicate a real-valued vector for each word. Different applications, such as information retrieval, document classification, question answering, named entity recognition, and parsing, consider these vectors as features. Most word vector methods attain distance or angle between pairs of word vectors to assess the word representation quality (Pennington et al., 2014). Vector space models (VSMs) perform well on tasks that involve measuring the similarity of meaning between words, phrases, and documents. Salton et al. (1975) focus on measuring document similarity, treating a query to a search engine as a pseudo-document. The relevance of a document to a query is given by the similarity of their vectors. Deerwester et al. (1990) show that we can focus on measuring word similarity, instead of document similarity, by looking at row vectors in the term-document matrix instead of column vectors. The distributional hypothesis in linguistics is that words that occur in similar contexts tend to have similar meanings. This hypothesis is the justification for applying the VSM to measuring word similarity. A word may be represented by a vector that derives vector ele-

ments from the word occurrences in various contexts. Similar row vectors in the co-occurrence matrix indicate similar word meanings (Turney & Pantel, 2010).

There are two methods for word representation: "count-based" representations, and "neural" or "prediction-based" embeddings. Explicit PPMI matrix, non-negative matrix factorization (NMF) (Wang & Zhang, 2012), or singular value decomposition (SVD) (Landauer & Dumais, 1997) of the co-occurrence matrix is count-based, and Skip-gram negative sampling (SGNS) (Mikolov et al., 2013a, 2013b), Global vectors for word representation (Glove) (Pennington et al., 2014), and FastText (Bojanowski et al., 2017), are "predictive" embeddings. The skip-gram with a negative-sampling training method (SGNS) was popularized via word2vec (Mikolov et al., 2013a; Mikolov et al., 2013b). Recent trends suggest that neural network-inspired word embedding models outperform traditional count-based distributional models on word similarity and analogy detection tasks. These models represent each word as a d dimensional vector of real numbers. Vectors that are close to each other are shown to be semantically related (Levy et al., 2015).

Nearly all such embedding methods produce dense representations for words whose coordinates in themselves have no meaningful interpretation. The numerical values of a word's embedding are interpretable only about other word representations. It is substantial to design an interpretable embedding whose coordinates have a distinct meaning to humans. Panigrahi et al. (2019) are reporting that there are multiple authors who have considered converting the existing embeddings to interpretable ones. Murphy et al. (2012) use non-negative matrix factorization of the co-occurrence matrix to derive interpretable word embeddings. Yogatama and Smith (2015) propose Sparse Overcomplete Word Vectors by solving an optimization problem in the dictionary learning setting. It produces a sparse non-negative high dimensional projection of word embeddings. Sun et al. (2016) use the CBOW model in their study and add the l_1 regularizer into its learning objective to generate interpretable sparse vectors. Subramanian et al. (2018) use a k-sparse denoising autoencoder to construct a sparse non-negative high dimensional projection of word embeddings, which they called SParse Interpretable Neural Embeddings (SPINE). Sparseness and non-negativity are desirable characteristics of word vectors that make them interpretable.

In general, the main idea of above-mentioned studies on the construction of interpretable word vectors using neural word embedding is to induce sparseness in the dimensions of word vectors. They have not attempted to elucidate the dimensions of dense word embeddings. Instead, they have learned sparse interpretable vectors with high dimensions (even 3000). They use projection vectors for word vector transformation and do not have a semantic category label to describe each dimension (Arora et al., 2018; Yogatama & Smith, 2015). These articles used the word intrusion test to quantify the interpretability. Reference (Park et al., 2017) applies the matrix rotation algorithm to get low-dimensional interpretable word vectors. But the resulting vectors cannot elucidate a conceptual label for each dimension. Jang and Myaeng (2017) try to identify conceptual property for each dimension of word embeddings using a categorization dataset called HyperLex. On average, there are a small number of words in the HyperLex dataset for each semantic category (2 words per category). Therefore, it cannot provide a comprehensive analysis of the word vectors dimensions. Şenel et al. (2018) produce a comprehensive dataset called SEMCAT. It transforms Glove vectors into interpretable word vectors that have 110 dimensions. Each dimension corresponds to a category word of SEMCAT. The word intrusion test needs human annotations and is an expensive evaluation method. So, the article by Park et al. (2017) quantifies interpretability using the distance ratio (DR) criterion. This method of evaluating interpretability does not represent human interpretations because it uses word vectors directly. Şenel et al. (2018) produce a comprehensive dataset called SEMCAT, with an average of 90 words per category. It also tries to evaluate the interpretable Glove vectors with 110 dimensions, using the interpretability score. Each dimension of the interpretable word vectors obtained in this method is equivalent to a category word. The main disadvantage of methods in references (Jang & Myaeng, 2017; Şenel et al.,

2018) is that the equivalent concept of each dimension of the resulting interpretable word vectors is strongly dependent on the category dataset and cannot find equivalents for each dimension using the corpus knowledge.

Although studies have made good progress on the generation of interpretable word vectors, there are still some drawbacks. No attempt been made to clarify a particular concept by a specific dimension using corpus knowledge. For example, in the paper by Şenel et al. (2018) Glove embeddings are transformed to a 110-dimensional space corresponding to SEMCAT dataset categories, but the interpretability score of the resulting interpretable word vectors is strongly dependent on the number of SEMCAT categories. But in the proposed method, we have introduced a basis word for each dimension, independent of the SEMCAT dataset, which presents the word vectors dimensions at the grain level. Also, in addition to constructing interpretable word vectors and finding the fine-grained equivalent of each dimension of the word vector, the proposed method introduces N of the most informative context words in the corpus as the final basis words. The extracted final basis words with higher rank can be used for applications such as keyword extraction and topic mining.

3. Original Contribution

We propose a framework that uses a new word selection method via the comparison of distance matrices to derive basis vectors for distributional representations. Distributional semantic vectors are high dimensional, and most researchers use fusion methods like SVD or NMF methods to reduce distributional vectors dimensions. We introduce a method to produce low dimensional word vector using the word selection concept. The main advantage of this method is constructing word vectors that are fully interpretable. It means that every basis vector of such word vectors is a meaningful basis word, while SVD or NMF produces word vectors with no interpretability. Word embedding methods such as word2vec, Glove, and so on have nothing to do with the interpretability of word vectors. To the best of our knowledge, this is the first approach that yields a semantic representation of words satisfying interpretability. Besides, in this framework, we have introduced a projection function $\mathcal{F}$. It derives the co-occurrence matrix of the vocabulary based on the localized co-occurrence. We describe this new framework in Section 5.

4. Dimensionality Reduction and Interpretability

Word vectors usually have high dimensions. Dimension reduction is significantly necessary to achieve higher efficiency in working with high dimensional semantic vectors. Reducing vectors to lower dimensionality is common in the construction of distributional semantic vectors. According to evidence, dimension reduction methods like NMF do not affect the quality of semantic vectors, and it maybe improves the quality of these vectors (Mikolov et al., 2013c). However, reduced vectors are not meaningful. This paper intends to introduce a method that is capable of producing some meaningful basis vectors while the efficiency degradation keeps low. NMF keeps all information; therefore, it provides good performance. The drawback of NMF is basis vectors with no possibility for meaningful interpretation. We show that you can obtain meaningful basis vectors while using a word selection method for controlling the information loss. In this research, we begin with 5K most frequent words as initial basis words. We introduce a word selection method for selecting N final basis words from the initial ba-

sis words. We determine N in a way that the Spearman's correlation coefficient does not degrade more than a given margin. Our experiments, which using MEN and Simlex-999 data sets, show we can reduce the dimensionality up to 1500 while the drop in Spearman's correlation coefficient performance is about 1.5 to 2%.

Despite NMF's considerable success and widespread adoption, a drawback of dimension reduction methods lies in their inability to provide a meaningful interpretation of the new dimensions. It is important to understand what exactly these dimensions signify. What kinds of properties are being obtained by these dimensions?

An interpretable word representation allows us to develop the word similarity measure that can justify why two words are similar. The first step to work on sentiment, concreteness, and frequency is a general decomposition of word vector spaces into meaningful, dense subspaces. Also, we need interpretable dimensions; because all information contained in a word vector does not help the model accurately attain the meaning of a word in the context (e.g., company-related senses of apple in fruit-implying context).

In this research, we introduce a word selection method. It selects N most important initial basis words with high ranks as final basis words. The N selected basis words represent interpretable basis vectors, and semantic vectors have meaningful dimensions. In contrast with NMF, which is a fusion method and semantic vectors dimensions produced with it have no interpretation. Intuitively, our method vectors have many zero values, so they inevitably carry less information, and we have found that this interpretability comes at a cost. However, it is possible to achieve good interpretability without a large penalty.

5. Proposed Framework

In this Section, we propose a framework for obtaining interpretable distributional semantic vectors using the new word selection method. In this framework, we use several steps to attain distributional semantic vectors that are meaningful, and each final basis word is equivalent to one context word. The operational steps of this framework are outlined below.

5.1. The Similarity between Two Target Words Using a Set of Basis Words

Corpus C contains M sentences that each word of a sentence has a POS tag. We arrange the words that are nouns, verbs, adjectives, and adverbs based on the number of events in the corpus. Then, we consider $\mathcal{N}$ words with higher frequency as the initial basis words. We refer to initial basis words set by BW. This set contains $\mathcal{N}$ initial basis words, which are defined as follows.

$$BW = [bw_1, bw_2, \dots, bw_{\mathcal{N}}] \qquad (5)$$

For each target word w_i, we consider the word w_i vector named $\overrightarrow{Vw_i}$ as follows. The number of components in $\overrightarrow{Vw_i}$ is equal to the number of initial basis words in the BW set.

$$\overrightarrow{Vw_i} = [vw_{i,1}, vw_{i,2}, \dots, vw_{i,\mathcal{N}}] \qquad (6)$$

Then, we use the cosine similarity measure to calculate the similarity of the target words w_i and w_j in the corpus C.

$$WS(w_i, w_j) = Cosine\ similarity(\overrightarrow{Vw_i}, \overrightarrow{Vw_j}) \qquad (7)$$

We use projection function $\mathcal{F}$ to obtain components of the vector $\overrightarrow{\mathrm{Vw_i}}$, Which indicates the degree of co-occurrence of the target word $\mathrm{w_i}$ and the basis word $\mathrm{bw_k}$.

$$vw_{i,k} = \mathcal{F}(w_i, bw_k) \qquad (8)$$

5.2. Projection Function $\mathcal{F}$

We introduce a projection function $\mathcal{F}$ to produce the real vector $\overrightarrow{Vw_i}$. The function relies on two quantities, namely localized co-occurrences between the target word w_i and the basis word bw_k, and a measure of association PPMI. Function $\mathcal{F}$ is introduced here. We denote the localized co-occurrences by LCO($\mathrm{w_i}$, $\mathrm{bw_k}$). It is an extension to co-occurrence number which, simply is the number of sentences in the corpus that $\mathrm{w_i}$ and $\mathrm{bw_k}$ co-occur. The localization method obtains the localized co-occurrence of the target word $\mathrm{w_i}$ and the basis word $\mathrm{bw_k}$.

To count the co-occurrence of the target word and the context word bw_k, bw_k can be in any situation within a sentence containing the target word. If we consider a window with a size of ± Win, the context word bw_k can be in the Win word before and after the target word situation. But if the context word bw_k is out of the window, it will not be counted. In this study, we present a localization method and assign a more significant weight to the context words closer to the target word. We also assign a smaller weight to words farther away from the target word. Sentences in the ukWaC corpus usually have a large number of words. We use the exponential coefficient $e^{-\alpha d_s(w_i, bw_k)}$ to count the co-occurrence. If the distance between the context word bw_k and the target word w_i is less, the exponential function considers a higher weight for the count, and the greater distance leads to a smaller weight. But unlike the window method, the word effect is not removed. When the number of words in a sentence is high, many words are not placed in the window.

We obtain localized co-occurrence as follows:

For all M sentences in the corpus:

1. Find sentences containing words w_i and bw_k.
2. Calculate $d_s(w_i, bw_k)$, which is the distance of the words w_i and bw_k in the sentence s.
3. Put LCO for sentence s equal to $\mathrm{e^{-\alpha d_s}}$
4. Add up to LCO's of previous sentences

So, localized co-occurrence counts of the basis word bw_k and target word w_i is defined by

$$\mathrm{LCO}(w_i\ , bw_k) = LCO_{i,k} = \sum_{i \epsilon M} e^{-\alpha d_s(w_i, bw_k)} \qquad (9)$$

Note that, $d_s(w_i, bw_k)$ is the distance of 2 words and there is no distinction between the left and right words. Higher $LCO_{i,k}$ means that the target word w_i and the basis word bw_k have co-occurred many times in the sentences of the corpus.

We attain the localized co-occurrence matrix based on Equation (9) for i=1, … $\mathcal{T}$ and k=1, …, $\mathcal{N}$. $\mathcal{T}$ is the total number of target words. The $\mathrm{i_{th}}$ row of the co-occurrence matrix is the vector $\overrightarrow{Vw_i}$, and the $\mathrm{k_{th}}$ column of the co-occurrence matrix refers to $\mathrm{k_{th}}$ basis word. Next, we replace each component

of the localized co-occurrence matrix by applying the PPMI method to the component. The resulting is our final co-occurrence matrix.

To construct the co-occurrence matrix, we count the vocabulary target words with initial basis words based on the following Equation:

$$\sum_{i=1}^{M} \sum_{All\ target\ words} \sum_{Context\ words} LCO_{i,k} = \sum_{i=1}^{M} \sum_{i=1}^{\mathcal{T}} \sum_{k=1}^{\mathcal{N}} e^{-\alpha d_s(w_i, b_{wk})} \quad (10)$$

M is the number of sentences in the corpus. $\mathcal{N}$ is the number of initial basis words, and $\mathcal{T}$ is the number of target words. The time complexity of the co-occurrence matrix construction is O($M \times \mathcal{T} \times \mathcal{N}$). Therefore, by reducing the number of context words from 5000 to 1500, the time complexity is reduced by 3.3 times. Because the number of sentences in corpus and the number of target words in the vocabulary is very large, the execution time is significantly reduced.

5.3. Refining Initial Basis Words (BWs)

In this section, we explain how we attain important words in initial basis words set. At first, we setup a vocabulary of target words based on the corpus C. The target words will be embedded using final basis words. The vocabulary is set up as follows.

1. Extract nonstop words from corpus C
2. Rank most frequent nouns (m.f.Noun)
3. Rank most frequent verbs (m.f.Verb)
4. Rank most frequent adjectives (m.f.adjectives)
5. Rank most frequent adverbs(m.f.adverbs)
6. Set vocabulary with $\begin{cases} m_N \text{ of m.f.Noun} \\ m_V \text{ of m.f.Verb} \\ m_{ADJ} \text{ of m.f.adjective} \\ m_{ADV} \text{ of m.f.adverbs} \end{cases}$

In the second step, we create a vector $\overrightarrow{Vw_i}$ for each target word w_i in the vocabulary. Component $vw_{i,k}$ of the target word vector $\overrightarrow{Vw_i}$ are obtained by applying the projection function $\mathcal{F}$ to a target word w_i and an initial basis word bw_k. The resulting target words vectors form the rows of the localized co-occurrence matrix. The number of columns of the localized co-occurrence matrix equals the number of the initial basis words. We call this localized co-occurrence matrix X. The matrix X is a $\mathcal{T} \times \mathcal{N}$ data matrix with $\mathcal{T}$ rows of target words that have $\mathcal{N}$ columns. Let FD(X) denotes the full distance matrix of X, which is symmetric and non-negative. Distance between the target words w_i and w_j in X is characterized by entry $FD(X)_{ij}$, which is calculated by the Euclidean distance of the vectors $\overrightarrow{Vw_i}$ and $\overrightarrow{Vw_j}$.

We examine the significance of k_{th} initial basis word by removing k_{th} column from the matrix X. The matrix X_{-k} denotes X in which the column k is removed. Then we calculate the distance matrix for the matrix X_{-k}, and call it the restricted distance matrix ($RD(X_{-k})$). Then, we obtain the restricted distance matrix for each initial basis word. Both full distance matrix and restricted distance matrix are

$\mathcal{T} \times \mathcal{T}$. In the following, we use the ranking algorithm to select some of the initial basis words as final basis words, which are important and effective.

We use a simple removal algorithm for basis word selection. The algorithm ranks the initial basis words using the following pseudo-code:

1) Calculate the full distance matrix FD(X).
2) For all initial basis words k=1, …, $\mathcal{N}$
   a) Calculate the distance matrix when k_{th} initial basis word is excluded (RD (X_{-k})).
   b) Calculate the difference between FD(X) and RD (X_{-k}) matrices.
   c) Calculate the Frobenius norm of the difference. The norm is a measure of importance of k_{th} initial basis word.
3) Rank the initial basis words based on the Frobenius norm. Higher norm gets a higher rank.

In step 2.c of the above pseudocode, we need to calculate the distance between two distance matrices. We use the Frobenius norm, which is one of the matrix norms also called the Euclidean norm. Frobenius norm calculates the 2-norm of the column vector. When we want to compute how close are two matrices A and B, we use $\|A - B\|_F$. The Frobenius norm of matrix A which is $\mathcal{T} \times \mathcal{T}$ is defined as follows (Yang & Shahabi, 2004):

$$\|A\|_F = \left(\sum_{i=1}^{\mathcal{T}} \sum_{j=1}^{\mathcal{T}} (a_{ij})^2\right)^{1/2} = (\text{trace}(A^T A))^{1/2} \qquad (11)$$

Hence, for each initial basis word k, we calculate the difference matrix (A= FD(X) − RD(X_{-k})), then we compute $\|A\|_F$.

Here, we consider $\mathcal{N}$=5K of the most frequent nouns, verbs, adjectives, and adverbs in corpus C as initial basis words. So, we calculate the Frobenius norm for all 5K initial basis words. We sort Frobenius norms of initial basis words in descending order because the big Frobenius norm means that the full distance matrix and the restricted distance matrix are more distanced.

6. Experimental Setup

We have studied extensively the processes that have high impacts on the efficiency of our approach. We have looked at the distance choices for calculating FD(X) and RD (X_{-k}) and in the meantime the choices for ranking calculation. We have examined various combinations of Euclidean distances and cosine similarities for matrix calculation on one hand, and Frobenius norm, Pearson correlation coefficient and R_V coefficient on the other hand. We report here that the best choice is Euclidean distance for the matrix calculation and the Frobenius norm for the ranking calculation.

6.1. Corpus

We use co-occurrence data from the Web-derived ukWaC corpus (http://wacky.sslmit.unibo.it/). The ukWaC is a very large English corpus that is built by web crawling. This corpus contains basic linguistic annotation (part-of-speech tagging and lemmatization) and it aims to serve as a general-

purpose resource for the English language. This corpus contains more than a billion words (Baroni et al., 2009). The ukWaC is among the largest corpora (Baroni et al., 2009), so we are using the first part of data that is named ukwac_dep_parsed_01 to resolve computational limits.

6.2. Construction of Localized Co-occurrence Matrix

We constitute the vocabulary with the top 20K most frequent noun lemmas, 10K verb lemmas, 10K adjective lemmas, and 5K adverb lemmas from the corpus. The 5K most frequent lemmas (nouns, verbs, adjectives, and adverbs) constitute the initial basis words (columns) of our co-occurrence matrix. For all target words, we extract localized co-occurrence counts of a target word in vocabulary with $\mathcal{N} = 5$K initial basis words. So we obtain localized co-occurrence matrix X. Each row of matrix X is a semantic vector.

6.3. PPMI

We know that raw word frequency is not a great measure of association between words. Positive Pointwise Mutual Information (PPMI) is a measure to determine that an initial basis word is particularly informative about target words or not. We transformed raw localized co-occurrence counts into Positive Pointwise Mutual Information (PPMI) scores.

6.4. Selection of N Final Basis Words as Basis Vectors

We used our word selection method to choose N final basis words as basis vectors from $\mathcal{N}$=5K initial basis words via comparison of distance matrices. We have applied the word selection method to two training sets. The first training set includes 18K target words. This set contains 8K most frequent nouns, 4K most frequent verbs, 4K most frequent adjectives, and 2K most frequent adverbs. As a result, we use a localized co-occurrence matrix Xt1 with 18K rows and $\mathcal{N}$=5K columns.

The second training set contains 12K target words. We deal with a localized co-occurrence matrix Xt2 with 12K target words and $\mathcal{N}$=5K initial basis words. Target words include 6K most frequent nouns, 3K most frequent verbs, 2K most frequent adjectives, and 1K most frequent adverbs. In the vocabulary, the ratio of nouns frequency to verbs frequency is 2, adjectives frequency to verbs frequency is 1, and adverbs frequency to verbs frequency is 0.5. Therefore, we almost maintain this ratio in training sets 1 and 2. It is noteworthy that in each word POS tag (noun, verb, adjective, and adverb), we select the most frequent ones.

In the first step, we calculate the original distance matrix for both Localized co-occurrence matrices Xt1 and Xt2. In the next step, we build the restricted distance matrix for each initial basis word by removing it from $\mathcal{N}$=5K initial basis words for matrices Xt1 and Xt2. Afterward, we calculate the Frobenius norm of the difference between the full distance matrix and the restricted distance matrix for the first and second training sets. Then, we find initial basis words whose absence in the basis words leads to a higher norm and less similarity between original distance matrix and restricted distance matrix. We choose N=500, 1000, 1500, 2000, and 3500 of initial basis words as final basis words that have more effect on the difference between two distance matrices. Therefore we have semantic vectors with N final basis words as basis vectors.

6.5. Word Similarity Experiments

In this Section, we investigate the effect of the word selection method via comparison of distance matrices on the quality of basis vectors using standard word similarity datasets. The datasets consist of word pairs and a gold standard score that indicates the human judgment of the similarity between the

words within each pair. We calculate the similarity between word vectors for each pair by the cosine similarity measure.

We measure cosine similarity on the MEN dataset (Bruni et al., 2012), SimLex-999 dataset (Hill et al., 2015), RG-65 datasets (Rubenstein & Goodenough, 1965), and WordSim353 dataset (Finkelstein et al., 2001). The MEN dataset includes similarity rates for 3000-word pairs. The SimLex-999 dataset has a 999-word pair's similarity rates. RG-65 dataset has 65-word pairs with their similarity. WordSim353 dataset contains 353-word pairs with their similarity rates. There are 45K vectors (20k nouns, 10k verbs, 10k adjectives, 5k nouns) in our vocabulary. Our vocabulary consists of all word pairs of datasets mentioned above. We compared the similarity of word pairs with a gold standard score through Spearman's correlation coefficient ρ.

7. Results and Discussion

In this section, we report the evaluation results of the proposed method using the word similarity task. Also, we examine the interpretability of word vectors quantitatively and qualitatively.

7.1. Word Similarity Task

First, we have an ablation study on the number of initial basis words (N) for constructing target word vectors. We generate the vocabulary target word vectors using N = 15000, N = 12000, N = 9000, N = 7000, N = 5000, and N = 3000 initial basis words. Then, we evaluate the vocabulary target word vectors using the word similarity task on the MEN, RG-65, SimLex-999, and WordSim353 test sets. The results of the evaluations are shown in Figure 1. As shown in Figure 1, by reducing the initial context words from 15000 to 7000, there is no drastic change in the Spearman's correlation coefficient in the MEN test set. In the case of the RG-65, SimLex-999, and WordSim353 test sets, the Spearman's correlation coefficient is increased slightly by decreasing the initial context words from 15,000 to 7,000. By reducing the initial context words from 7000 to 5000, the Spearman's correlation coefficient of the MEN, RG-65, SimLex-999, and WordSim353 test sets is increased by 0.21%, 1.85%, 0.22%, and 0.44%, respectively. As shown in Figure 1, by reducing the initial context words from 5000 to 3000 the Spearman correlation coefficient in the MEN, SimLex-999, and WordSim353 test sets is decreased by about 2%. The decrease in Spearman's correlation coefficient in the RG-65 test set is more severe by 4.6%. So, we select N=5000 initial context words. According to our studies, the RG-65 test set shows a sharp decrease or increase in Spearman's correlation coefficient by changing the factors of methods compared to other test sets due to the small number of word pairs (65-word pairs).

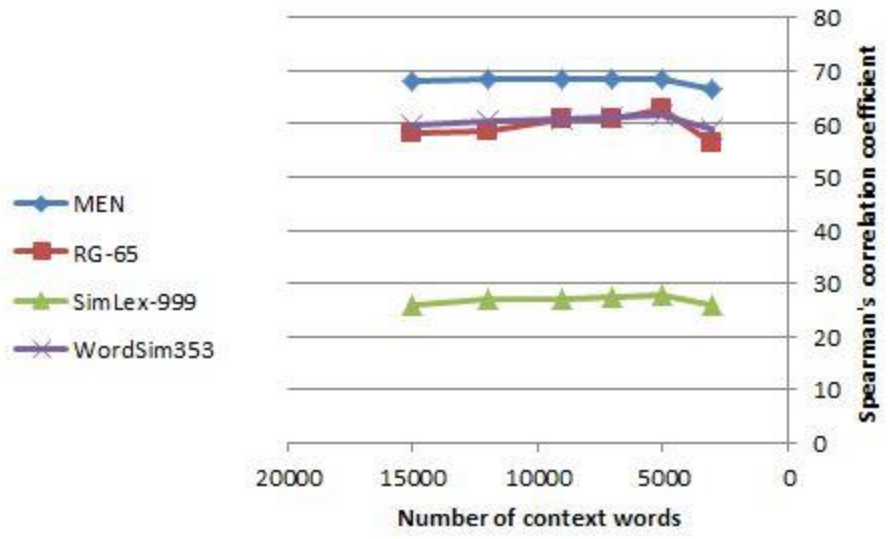

Figure 1: An ablation study on the number of initial context words

Then, we report the Spearman's correlation coefficient of co-occurrence matrix in three cases. One is "using localization", the other is "no-localization and window=10", and third one is "no-localization and no-window". Table 2 reports the resulting data. Results show that the co-occurrence matrix X is more informative when we apply the localization compare to no-localization is applied. We report that there are improvements in the Spearman's correlation coefficient by 1.8%, 5.7%, 1.4%, and 3.5% for MEN, RG-65, Simlex-999, and WordSim353 datasets respectively. Also, the co-occurrence matrix X using localization is more effective than the matrix X using no-localization and no-window. Localization improves Spearman's correlation coefficient about 4.5%, 9%, 4.3%, and 6% for MEN, RG-65, Simlex-999, and WordSim353 datasets respectively compare to no-localization and no-window. Therefore, localization method is so effective. We set $\alpha = 0.1$ in these experiments. This comes from our judgment on the decaying effect of α.

	using localization	**no- localization, win=10**	**no-localization and no-window**
MEN Dataset	68.62	66.89	64.03
RG-65 Dataset	62.70	56.96	53.48
SimLex-999 Dataset	27.66	26.22	23.27
WordSim353 Dataset	61.66	58.16	55.48

Table 2: Spearman's correlation coefficient (ρ) for co-occurrence matrix using localization, using no-localization and window=10, and using no-localization and no-window.

We have done all coming experiments and results using "localization" and alpha=0.1. Then, we use the word selection method to choose N final basis words from $\mathcal{N}$=5K initial basis words of localized co-occurrence matrix as basis vectors. We obtain vocabulary vectors with N=500, 1000, 1500, 2000, and 3500 basis vectors to examine the effect of reducing the number of basis words on the quality of the target word vectors. Then, we compute the cosine similarity of word pairs that are present in

vocabulary. We evaluate the proposed word selection method using the gold standard scores of MEN, SimLex-999, RG-65, and WordSim353 datasets. Tables 3 and 4 represent the Spearman's correlation coefficient (ρ) between cosine similarity and a gold standard score of word pairs for training sets 1 and 2, respectively.

Table 3 shows the results when the first training set of 18K target words is considered. The word selection method refines (reduces) $\mathcal{N}$=5K initial basis words into N selected basis words as basis vectors. Table 3 reports the Spearman's correlation coefficient of the word selection method using N=500, 1000, 1500, 2000, and 3500 selected basis words. The results show a slight decrease in accuracy by reducing initial basis words from 5K to 3500. Accuracy decreases by 0.58%, 0.7%, and 0.23% for MEN, RG-65, and simLex-999 datasets respectively. Accuracy increases by 0.64% for the WordSim353 dataset using 3500 selected basis words. Vocabulary vectors using 2000 selected basis words reduce accuracy by about 1.34%, 0.84%, and 0.21% for MEN, RG-65, and simLex-999 datasets, respectively. It increases the WordSim353 dataset accuracy by 0.16%.

Target words vectors using 1500 selected basis words reduces Spearman's correlation coefficient 1.66%, 1.9%, 0.23%, and 0.37% for MEN, RG-65, simLex-999, and WordSim353 datasets, respectively. Results in Table 3 represents that 1000 selected basis words reduces Spearman's correlation coefficient 2.36%, 2.98, 2.07, and 2.13% for MEN, RG-65, simLex-999, and WordSim353 datasets, respectively.

By reducing the number of basis words to 500, the Spearman's correlation coefficient severely reduces for MEN, RG-65, and Simlex-999 datasets. There is a 6.37% decrease in accuracy for the MEN dataset and a 4.1% decrease for the RG-65 dataset. Also, the Spearman's correlation coefficient decreases by 2.07% and 1.43% for the SimLex-999 and the WordSim353 datasets, respectively.

The results of Table 3 show that by reducing the number of basis words from 5000 to 1500, the accuracy reduction is below 2%. By reducing the number of basis words to 1000, the accuracy decreases by about 2 to 3%. It seems that the word selection method can reasonably reduce the number of initial basis words from 5K to 1500 while there is less than 2% of accuracy reduction.

	Initial basis words	**selected basis words**				
Number of basis words	**5k**	**500**	**1000**	**1500**	**2000**	**3500**
MEN Dataset	68.62	62.25	66.26	66.96	67.28	68.04
RG-65 Dataset	62.70	58.60	59.72	60.80	61.86	62.00
SimLex-999 Dataset	27.66	25.59	26.72	27.13	27.45	27.43
WordSim353 Dataset	61.66	60.23	59.53	61.29	61.82	62.30

Table 3: Spearman's correlation coefficient (ρ) for the word selection method with N selected basis words on the first training set.

Next, we try to investigate the effect of the number of target words in the training set on the accuracy obtained from reducing the number of basis words. We apply the method on the second training set. The second training set has 12K target words. Table 4 presents the results on the second training set. The word selection method refines $\mathcal{N}$=5K initial basis words to N final basis words as basis vectors. Results presented in Table 4 show that by selecting 3500 important basis words, we just have a 0.7%, 1.2%, and 0.33% accuracy drop for MEN, RG-65, and simLex-999 datasets, respectively. It slightly improves accuracy for WordSim353 dataset by 0.06%.

We observe 1.13%, 1.63%, and 0.71% accuracy drop for MEN, RG-65, and simLex-999 datasets by 2000 selected basis words, respectively. Our method with 2000 basis words increases the Spearman's correlation coefficient by 0.22% for the WordSim353 dataset. Spearman's correlation coefficient for vectors with 1500 selected basis words as basis vectors reduces 1.19%, 1.64%, and 0.29% for MEN, RG-65, and simLex-999 datasets, respectively. Also, WordSim353 dataset accuracy increases by 0.85%.

By selecting 1000 basis words, we see 2.3%, 1.77%, 1.19%, and 0.7% decrease in MEN, RG-65, and simLex-999, and WordSim353 datasets, respectively. Target word vectors using 500 basis words reduces Spearman's correlation coefficient 2.25%, 4.08%, 2.54%, and 2.04% for MEN, RG-65, simLex-999, and WordSim353 datasets, respectively. We observe that there is not a big difference in word selection method accuracies by selecting 1500 basis words (about 1.5-2%). It means that the word selection method chooses the most valuable basis words and is efficient.

We like to report an important conclusion in this step. The training set is all information we use to refine the initial basis words. We investigate the accuracy when 12K and 18k words are used for this refinery. Tables 3 and 4 clearly show that the accuracy is not degraded while we reduced the number of target words from 18k to 12k.

	Initial basis words	**selected basis words**				
Number of basis words	**5k**	**500**	**1000**	**1500**	**2000**	**3500**
MEN Dataset	68.62	66.37	66.59	67.43	67.49	67.92
RG-65 Dataset	62.70	58.62	60.93	60.04	61.07	61.50
SimLex-999 Dataset	27.66	25.12	26.47	27.37	26.95	27.33
WordSim353 Dataset	61.66	59.62	60.96	62.51	61.88	61.72

Table 4: Spearman's correlation coefficient (ρ) for the word selection method with N selected basis words on the second training set.

Furthermore, using either first or second training sets show that we can refine $\mathcal{N}$=5K initial basis words to N=1500 final basis words with a little bit of efficiency loss. It decreases Spearman's correlation coefficient by a margin of 1.5-2% on gold test datasets. So we conclude that the word selection method based on the distance matrix is able to provide an acceptable result by selecting only 1500 basis words out of 5000 and is successful and efficient. Studies on the two training sets 1 and 2 shows that the number of target words in the training set does not have much effect on the accuracy obtained on gold test sets, and reducing the number of words from 18K to 12K does not cause a significant accuracy drop.

Next, we study the effect of the basis words reduction on the accuracy using fusion methods such as NMF. NMF tries to present all existing information in a compact form and in a smaller dimension. Each component of an NMF vector can represent several basis words while it is not interpretable. We construct the vocabulary vectors with 1000 basis vectors extracted by the NMF method. That means we convert the 45k × 5K localized co-occurrence matrix to a 45K × 1K matrix using the NMF method. We also transform the 45K × 3500, 45K × 2000, and 45K × 1500 co-occurrence matrices to the 45K × 1000 matrix by the NMF method. Table 5 reports the evaluation results of the vocabulary vectors on the gold test sets.

	Apply NMF on word vectors with						
Number of basis words	**1500 BW**		**2000 BW**		**3500 BW**		**5K BW**
Training set	1	2	1	2	1	2	**vocabulary**
MEN Dataset	69.25	69.71	70.47	69.96	72.47	72.62	73.69
RG-65 Dataset	61.34	60.75	64.61	64.19	68.88	67.18	71.62
SimLex-999 Dataset	27.52	27.42	28.22	28.04	29.31	29.57	30.87
WordSim353 Dataset	64.02	64.67	64.81	65.45	67.95	67.26	67.74

Table 5: Spearman's correlation coefficient (ρ) for 1000 dimensions by the NMF method

Table 5 shows the accuracy of vocabulary vectors with 1000 dimensions obtained by NMF on training sets 1 and 2. Also, it shows the accuracy obtained by reducing the dimensions of the localized co-occurrence matrix from 5000 columns to 1000 new columns. Comparing the results of NMF method on 5000 dimension vectors with 3500 dimension vectors shows 1%, 3%, 1%, and 0.5% accuracy drop in MEN, RG-65, simLex-999, and WordSim353 datasets, respectively.

Applying the NMF method to the co-occurrence matrix using 2000 basis words decreases accuracy about 3% in the MEN dataset, and 2.5% in simLex-999, and 2.5% in WordSim353 datasets. Also, a 7% decrease in the RG-65 dataset accuracy has occurred, which is due to the small number of words in the RG-65 dataset. For vocabulary vectors with 1,500 basis words, the Spearman's correlation coefficient decreases 4% for the MEN dataset, and 3% in simLex-999, and 3% in WordSim353 datasets. Results show a 10% decrease in the Spearman's correlation coefficient in the RG-65 dataset. It is mainly due to the small number of words in the RG-65 dataset and also small number of basis words that is 1500. Gold datasets with a large number of word pairs such as MEN and simLex-999 experience 3% to 4% decrease in Spearman's correlation coefficient. Note that we have used only 1500 basis words. It means by reducing the number of basis words from 5K to 1500, the NMF method reduces accuracy for 1000 dimensions by about 3-4%. We conclude that the performance of our selection method and NMF method drop when a fewer number of basis words are used. Figure 2 shows the decreasing trend of the Spearman's correlation coefficient by reducing the number of basis words from 5000 to N = 3500, 2000, 1500 basis words as in the training set 1. We report the Spearman's correlation coefficient on gold test sets. We also apply the NMF method to matrices with N basis words and obtain 1000 dimensions. We observe the decreasing trend again. The accuracy reduction in the word selection method is less than the NMF method. The NMF performance is decaying faster in the RG-65 dataset.

Figure 3 shows the decrement diagram of the Spearman's correlation coefficient for the word selection method and the NMF method obtained using Training set 2. You can see that the results are very similar to Figure 2. This similarity indicates that the result of the word selection method does not depend on a large number of target words in the training set.

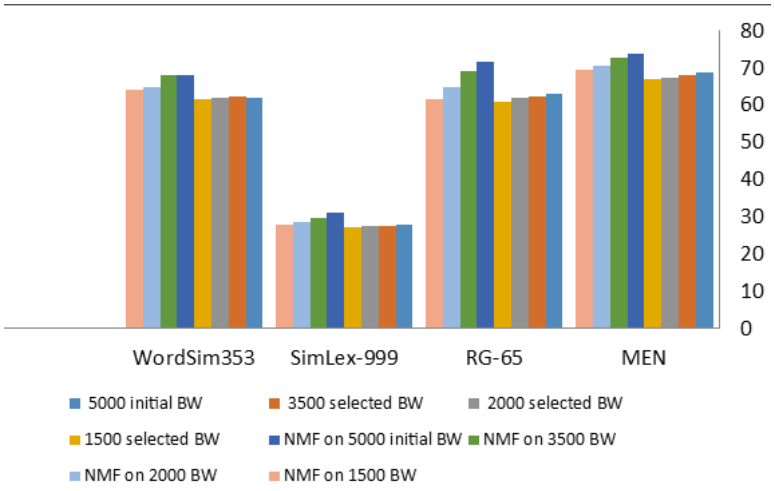

Figure 2: Decreasing trend of Spearman's correlation coefficient for word selection method and NMF method using training set 1.

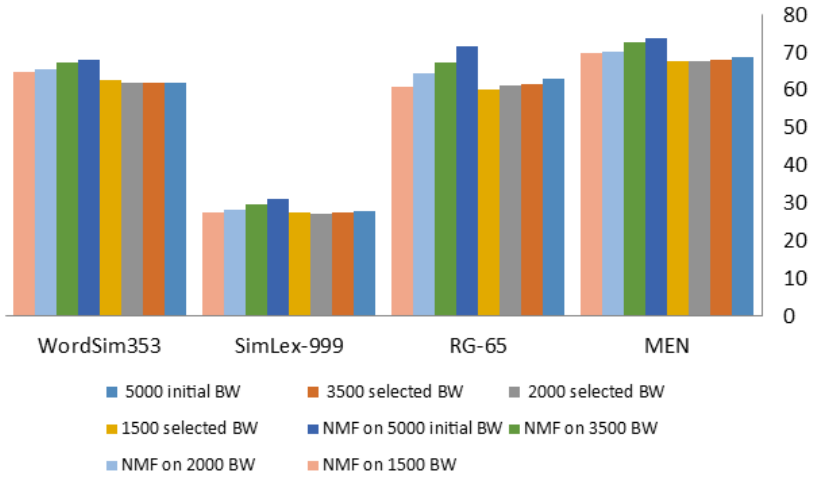

Figure 3: Decreasing trend of Spearman's correlation coefficient for word selection method and NMF method using training set 2.

In the selection method, we just select a subset of the initial basis words set, without any manipulation of data, so we need to find and remove the basis words which do not help in discrimination. But, fusion methods (like NMF) transform the dimensions to some other spaces to exploit the discrimination capability in the transformed space. Note that, we do not expect better performance for the selection method compares to the fusion methods, which use all information while applying the transformation. In this research, we have tried to introduce a selection method for producing meaningful basis words and keep the performance as high as possible. It is clear that any selection method cannot be better than the fusion method.

In this article, we present a word selection method that selects N basis words from initial basis words set with minimal loss of accuracy. Each basis vector corresponds to a basis word, and the resulting word vectors are meaningful and interpretable. To the best of our knowledge, this research is the first research on finding the basis vectors applying the word selection method to deduct interpretable word vectors. The results presented here show that our proposed selection method for N=1500 basis vectors has good accuracies; about 1.5-2 % lower than 5K initial basis words. The main advantage of our selection method is that each basis vector is equivalent to a basis word and is meaningful.

We obtained word embeddings by word2vec (Mikolov et al, 2013a), FastText (Bojanowski et al., 2017), and BERT (Devlin et al., 2018) using bert-base-nli-mean-tokens and stsb-roberta-large methods from the sentence_transformers library (Reimers & Gurevych, 2019). We evaluate the word embeddings using the word similarity task on the MEN, RG-65, SimLex-999, and WordSim353 test sets. The results are reported in Table 6 below.

	word2vec	**FastText**	**bert-base-nli-mean-tokens**	**stsb-roberta-large**	**Baseline Matrix**
Number of Dimensions	1000	300	768	1024	5000
MEN Dataset	76.3	75.39	59.56	51.41	68.62
RG-65 Dataset	72.9	77.67	77.40	53.21	62.70
SimLex-999 Dataset	41.83	44.04	60.92	61.77	27.66
WordSim353 Dataset	70.8	68.096	27.50	25.58	61.66

Table 6: Spearman's correlation coefficient (ρ) for word embeddings.

Table 6 shows that the word embeddings obtained by the word2vec and FastText models have larger Spearman's correlation coefficients than the count-based Baseline model. In the bert-base-nli-mean-tokens and stsb-roberta-large methods, which have 768 and 1024 dimensions, respectively, the Spearman's correlation coefficients on the MEN, RG-65, and WordSim353 datasets are lower than the other methods. However, in the SimLex-999 similarity set, a significant Spearman's correlation coefficient of 60% has been reported.

7.2. Qualitative Evaluation of Interpretability

The word embeddings obtained in neural methods are very compact vectors that are difficult for humans to interpret. We often do not know what the high value compared to the low value in one dimension determines. The main idea of qualitative evaluation is that if a particular dimension of the word vector be interpreted, the high-ranked target words of the vocabulary for that dimension must have semantic coherence (Jha et al., 2018). If a vector dimension is interpretable, high-ranking words for that dimension should display semantic or syntactic groupings. To confirm this qualitatively, we analyze word vectors for a few target words in vocabulary namely, "service", "speed", and "waterproof". These target words are randomly selected from the vocabulary. Then, we put the top 15 words in that dimension as the top-ranked words in Table 7. We see that the proposed model using 1500 dimensions compared to other models forms a coherent semantic group. Semantic coherence is not found in the top words of FastText, word2vec, and NMF models. In word vectors obtained by count-base

methods, each dimension is equivalent to one word. The Baseline model has 5K dimensions, which is equal to one of the most frequent words. We compare top words groups of the Baseline model for words "service", "speed", and "waterproof". We find that the top word groups in the Baseline model have general meanings. This is because the criterion for selecting initial basis words for dimensions is the most frequent words. The Baseline model compared to the FastText, word2vec, and NMF models creates a more coherent semantic group. The proposed model has 1500 dimensions. Each dimension corresponds to a word that is selected based on the word selection method. By examining and comparing the top words in Table 7, we found that the proposed model has formed a more coherent semantic group than the Baseline model for the target words (service, speed, waterproof). Note that words such as provide, information, support, public, customer, offer, which are closely related to the word "service", are among the top words for the target word "service". Also, in the semantic group for the word "speed", words with a very high association such as server, use, web, file, client, run, windows, system, network, access, user, and sql are seen. Also in the semantic group obtained for the target word "waterproof", words such as Jacket, wear, leather, trouser, dress, fleece, shirt, and pocket can be seen. By examining this semantic group, we can deduce that the corpus explains the waterproof property and clothing. According to Table 7, we see that this level of semantic grouping is not observed in any of the other models.

Model	# of Dims	Target words	Top words
FastText	300	service	vu, nm, mg, kg, kb, hz, mb, km, md, ml, mm, cm, pm, vt, lb
		speed	2f, sq, en, on, om, du, we, uc, ja, wp, je, vu, nl, bo, jp
		waterproof	8d, 4d, 1d, 5d, sj, 6d, p1, fm, 1c, 2d, rn, va, d., 1x, cv
word2vev	1000	Service	bupa, 1x, atmospheric, gprs, semester, giants, edina, abebooks, ovid, camel, pga, tpo, prise, consignment, riba
		Speed	sitemap, abingdon, mw, yn, ht, dealership, font-family, kg, lotus, reserved, pushchair, enhancing, abattoir, cod, radiator
		Waterproof	bonnet, uktv, cask, mm, affiliated, mce, properties, tsb, handmade, kinase, filler, cordless, ferries, condolence, creamy
NMF	1000	Service	heavy-duty, quantize, initial, sweaty, woolly, peter, outdoors, sinful, terracotta, insolent, last-minute, kilo, quicker, heady, golf
		Speed	Workers, deft, unrhyw, lift, re-ignited, fall, municipal, whig, bruise, funnel, reframing, ml, decommission, proletarian, spot-check
		Waterproof	fair, wright, creative, zanzibar, Uganda, counterfeit, negotiating, musing, wt, self-fertilised, plaintive, Kashmir, Sherlock, chas, padding
Baseline	5K	Service	be, provide, information, use, health, have, support, other, not, public, new, people, include, customer, offer
		Speed	be, high, limit, use, time, not, camera, have, more, road, do, up, new, increase, make
		Waterproof	be, not, do, have, use, more, good, so, also, make, only, work , time, other, now
Proposed	1500	Service	Server, be , use, web, not, file, client, run, windows, system, network, access, user, sql, other
		Speed	Wind, be, farm, energy, turbine, power, blow, not, have, strong, high, more, rain, make, day
		Waterproof	Jacket, be, wear, leather, trouser, more, make, black, not, dress, have , man , fleece, shirt, pocket

Table 7- Qualitative evaluation of the word vectors. We examine the high-ranking dimensions for three randomly selected words.

7.3. Quantitative Evaluation of Interpretability

One of the advantages of increasing interpretability is that each dimension can be understood to some extent by humans, and each dimension is associated with a recognizable concept. Quantitative evaluation of interpretability based on human judgments is an effective method. The most common measure of quantitative interpretability for a set of word embeddings is the word intrusion test introduced by

Chang et al. (2009). In each test, it produces five words that four words are related to, and one word is different from the other words. If human judgment recognizes the word influence well, the model is considered quantitatively interpretable. This method requires a human vote, so word intrusion test is an expensive method (Trifonov et al., 2018). In addition, the word intrusion test does not quantify levels of interpretability but determines interpretability using a binary decision. Continuous quantification of interpretability is more appropriate than binary decision because the levels of interpretability vary in different dimensions (Park et al., 2017; Şenel et al., 2018). In this paper, we quantify the word vector's interpretability using topic coherence and interpretability score measures.

7.3.1. TOPIC COHERENCE

Topic coherence is an automated evaluation method for interpreting topic models that are well related to human evaluations .Topic coherence is the mean pairwise similarity of word pairs. The mean coherence of all topics is called total topic coherence. Assume that $x_d^{(p)}$ is a word that has the rank p in the d_{th} dimension of word vectors. The coherence of d_{th} dimension is calculated based on the following Equation (Trifonov et al., 2018):

$$coh_*(d) = \frac{2}{n \,.\, (n-1)} \sum_{p=1}^{n-1} \sum_{q=p+1}^{n} sim_* \left(x_d^{(p)}, x_d^{(q)} \right) \qquad (13)$$

Parameter n is the number of high-rank words in each dimension that are used to calculate topic coherence .In this study, similar to reference (Trifonov et al., 2018), we consider n = 10. We compute Equation (13) for all pairs of words in the d_{th} dimension. Then we calculate the total topic coherence based on the following Equation (Trifonov et al., 2018):

$$coh_*(1, \dots, D) = \frac{1}{D} \sum_{d} coh_*(d) \qquad (14)$$

We use the cosine similarity criterion to measure sim_*. We calculate the total topic coherence on the models word2vec, FastText, Baseline, and the proposed model using N=1500 and N=1000 dimensions. The total topic coherence of the models is reported in Table 8. As shown in Table 8, the topic coherence of word2vec and FastText methods is much less than other methods .The Baseline method uses 5kmost frequent words as initial basis words to construct the co-occurrence matrix. The topic coherence of the Baseline method is higher than the word2vec and FastText methods by 0.1. Topic coherence of the proposed method using N=1500 final basis words compared to the Baseline method is increased by 0.12. By reducing the word vectors dimensions to 1000 using the proposed method, interpretability is increased by 0.22 compared to the Baseline method. That is, the word selection method selects the final basis words in such a way that the resulting low-dimensional word vector's interpretability is increased. Since in the proposed method using N = 1500 compared to N = 1000, the Spearman correlation coefficient is reduced to a lesser extent for the word similarity task; we recommend using N=1500 final basis words.

Model	# of Dims	$coh_*(\mathbf{1}, \dots, \boldsymbol{D})$
word2vec	1000	0.2013
FastText	300	0.2187
Baseline	5K	0.3111
Proposed	1500	0.4318
Proposed	1000	0.5381

Table 8: The total topic coherence of the models

7.3.2. INTERPRETABILITY SCORE

Reference (Şenel et al., 2018) suggests an alternative method for word intrusion test that achieves quantitative evaluation automatically and continuously which, is based on human judgment .This method uses category theory to study the semantic structure of word vector spaces. This requires categories that represent a wide range of distinct concepts and distinct types of relationships. To achieve this goal, reference (Şenel et al., 2018) has introduced a new dataset called SEMCAT. This dataset contains more than 6500 words, and the dataset are semantically classified into 110 categories. This research is based on the key idea that if a dataset shows all the groups a human makes up, one do not need to rely on human judgments. As a result, it is easy to check the presence or absence of distinct word embeddings dimensions in each of the dataset category. Therefore, a dataset with a sufficient number of categories can provide a good approximation for human judgments.

$$IS_{i,j}^{+} = \frac{\left|S_j \cap v_i^{+}(\lambda \times n_j)\right|}{n_j} \times 100 \qquad (15)$$

And

$$IS_{i,j}^{-} = \frac{\left|S_j \cap v_i^{-}(\lambda \times n_j)\right|}{n_j} \times 100 \qquad (16)$$

Where $IS_{i,j}^{+}$ and $IS_{i,j}^{-}$ show the interpretability score for the positive direction of the i_{th} dimension and the negative direction of the i_{th} dimension, respectively .Where, $i \in \{1,2,\dots,D\}$ and $j \in \{1,2,\dots,k\}$. Parameter D is the number of word vector dimensions and k is the number of categories in the dataset. S_j is the collection of words in the j_{th} category. The parameter n_j indicates the number of words in the j_{th} category. $v_i^{+}(\lambda \times n_j)$ is a set of distinct words with a high rank in the i_{th} dimension, and $v_i^{-}(\lambda \times n_j)$ is a set of distinct words with a low rank in the i_{th} dimension. $\lambda \times n_j$ specifies the number of words with high and low ranks. The parameter λ determines the strictness of the interpretability score. The variable λ can be changed in the range of 1 to 10. We consider the parameter $\lambda = 5$ similar to the reference (Şenel et al., 2018) to perform the experiments. Next, we calculate the interpretability score (IS) for the i_{th} dimension and the j_{th} category as follows:

$$IS_{i,j} = \max\left(IS_{i,j}^{+}, IS_{i,j}^{-}\right) \qquad (17)$$

Then, we calculate the interpretability score of the i_{th} dimension for all 110 categories by Equation (18):

$$IS_i = \max_j IS_{i,j} \qquad (18)$$

The total interpretability score of the word vectors for a model is calculated based on the following Equation:

$$IS = \frac{1}{D}\sum_{i=1}^{D} IS_i \qquad (20)$$

We obtain and report the interpretability score of word vectors obtained by FastText, word2vec, Baseline models, and proposed method using N=1500 and N=1000 . Interpretability score measurements are based on the SEMCAT dataset presented in reference (Şenel et al., 2018). Interpretability scores for word vectors derived from each model are calculated and reported in the table 9. As you can see in Table 9, the interpretability score of the proposed method is much higher than the neural models. Method τ^* presented in the paper by Şenel et al. (2018) transforms the obtained vectors of the Glove method to an interpretable space with 110 dimensions. In this method, the label of each dimension is one of the SEMCAT categories. The interpretability score of the τ^* method is also reported in Table 9. The interpretability score of the proposed method using N = 1500 compared to the τ* method has improved by 0.68.

Model	# of Dims	Interpretability Score
word2vec	1000	27.26
FastText	300	28.97
Baseline	5k	52.37
Proposed	1500	52.18
Proposed	1000	52.66
$\boldsymbol{\tau}^*$(Şenel et al., 2018)	110	51.5

Table 9: The interpretability score of word vectors obtained by different models

Examining the qualitative interpretability presented in Table 7 and the quantitative interpretability results presented in Tables 8 and 9, we find that the proposed method using N=1500 dimensions is more interpretable than word2vec, and FastText methods .The proposed method using N = 1500 reduces 3500 dimensions of word vectors. So the word selection method compared to the Baseline method decreases the interpretability score only slightly, which is justified by the 3.3-fold reduction of the word vector dimensions. The interpretability score of the proposed method using N=1000 compared to N=1500 is 0.48 higher. This result has already been obtained in Table 8 using the topic coherence method. But in the word similarity task, the accuracy drop for N=1000 is more than N=1500. For this reason, we recommend N=1500 for the number of final context words.

Also, the interpretability score of the proposed method is improved by N=1500 compared to the τ* method, which transforms the Glove vectors to the 110-dimensional interpretable space, and each dimension corresponds to a SEMCAT category. Therefore, although the proposed method in the word similarity task obtains a lower Spearman's correlation coefficient than the word2vec and FastText methods, it works better in terms of interpretability. The sparse vector has a large number of zeroes compared to the dense vector and inevitably carries less information. In the proposed method, compared to the Baseline method, by increasing the interpretability, there is a slight decrease in Spearman's correlation coefficient in the word similarity task, which is not a large penalty. In addition to improving the interpretability of word vectors, the proposed method introduces N final basis words

that represent the knowledge extracted from the corpus at a granular level. The methods that transform neural vectors to an interpretable space often fail to provide a concept for each word embedding dimension. In methods such as τ*, a category is assigned to each word embedding dimension, but these conceptual equivalents of each word embedding dimension depend on another dataset, such as SEMCAT. For example, if the corpus contains specialized medical information, for interpreting the embeddings using the method of reference (Şenel et al., 2018), we need to have a data set of categories in the medical field. However, in the proposed method, word vectors can be constructed simply by using a specialized corpus, that for each dimension of word vectors, a concept has been extracted from the corpus by the proposed method. Also, the best final basis words selected by the proposed method can be used in applications such as keyword extraction and topic modeling.

8. Conclusion

In this research, we introduce a selection method based on the comparison of distance matrices to produce meaningful basis vectors. Each basis vector corresponds to a basis word. We use vocabulary that contains 45K target words. We construct localized co-occurrence matrix X for target words in vocabulary using 5k initial basis words. We use the matrix X to get the similarity of word pairs in the test sets. We apply the word selection method on training sets 1 and 2 with different sizes to select N basis words. The results show that the selection method does not depend on increasing or decreasing the number of target words in the training set. It reports almost the same accuracy. We produce word vectors with N=500, 1000, 1500, 2000, and 3500 informative basis vectors. Our experiences show that reducing the number of basis vectors from 5000 to 1500 reduces the Spearman's correlation coefficient 1.5-2%. Note that basis words reduction decreases the accuracy not only in the selection method but also in the NMF method. Our comparative study shows that the accuracy of the selection method is less than the word2vec, FastText, and NMF method. The accuracy shortcoming of the selection method is compensated by the fact that we produce interpretable basis vectors corresponding to unique basis words.

By qualitatively evaluating the three target word vectors selected randomly from the vocabulary, we show that the top words obtained for these target words form more coherent semantic groups than other methods word2vec, FastText, Baseline. We use two automated evaluating methods, namely topic coherence and interpretability score, to quantitatively obtain the interpretability of word vectors obtained by word2vec, FastText, Baseline, and Proposed methods using N = 1500 and N = 1000. The results show that in the proposed method, the criteria of topic coherence and interpretability score are higher than neural methods. The proposed method word vectors compared to interpretable τ* word vectors, the interpretability score is slightly higher. Therefore, the proposed method selects N=1500 final basis words, and there is a slight decrease in accuracy in the word similarity task compared to the Baseline method. However, the interpretability score and topic coherence of the word vectors obtained by the proposed method have increased. In addition to increasing interpretability, the proposed method obtains N final basis words and extracts a specific concept for each dimension of the word vector at the granular level and does not require specialized external knowledge.

Here, we report the top 15 basis words of 1500 selected basis words. We like to draw your attention to the fact that these are the top 15 words based on a web crawled dataset in 2009. The words "God-disease-cell-loan-cancer-blood-oil-bird-fish-gas-Iraq-pain-beach-bedroom-christ" are selected using the word selection method. We can release the 1500 selected basis words for interested researchers to use. Note that, by applying this method to the favorite corpora, readers will be able to create their own vector space based on meaningful words. Then, they can develop different applications

such as information retrieval, document classification, question answering, and topic mining, and so on.

Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

Arora, S., Li, Y., Liang, Y., Ma, T., & Risteski, A. (2018). Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association for Computational Linguistics*, *6*, 483-495.

Bruni, E., Boleda, G., Baroni, M., & Tran, N. K. (2012, July). Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 136-145).

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, *5*, 135-146.

Baroni, M., Bernardini, S., Ferraresi, A., & Zanchetta, E. (2009). The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, *43*(3), 209-226.

Clark, S. (2015). Vector space models of lexical meaning. *The Handbook of Contemporary semantic theory*, 493-522.

Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., & Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems* (pp. 288-296).

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2001, April). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web* (pp. 406-414).

Grefenstette, E. (2013). Category-theoretic quantitative compositional distributional models of natural language semantics. *arXiv preprint arXiv:1311.1539*.

Heunen, C., Sadrzadeh, M., & Grefenstette, E. (Eds.). (2013). *Quantum physics and linguistics: a compositional, diagrammatic discourse*. Oxford University Press.

Hill, F., Reichart, R., & Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, *41*(4), 665-695.

Jang, K. R., & Myaeng, S. H. (2017, April). Elucidating conceptual properties from word embeddings. In *Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications* (pp. 91-95).

Jurafsky, D. and Martin, J.H., (2014). Speech and language processing (Vol. 3). London: Pearson.

Jha, K., Wang, Y., Xun, G., & Zhang, A. (2018, November). Interpretable word embeddings for medical domain. In *2018 IEEE international conference on data mining (ICDM)* (pp. 1061-1066). IEEE.

Kartsaklis, D. (2014). Compositional operators in distributional semantics. *Springer Science Reviews*, *2*(1-2), 161-177.

Levy, O., Goldberg, Y., & Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the association for computational linguistics*, *3*, 211-225.

Lenci, A. (2018). Distributional models of word meaning. *Annual review of Linguistics*, *4*, 151-171.

Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, *104*(2), 211.

Murphy, B., Talukdar, P., & Mitchell, T. (2012, December). Learning effective and interpretable semantic models using non-negative sparse embedding. In *Proceedings of COLING 2012* (pp. 1933-1950).

Mikolov, T., Yih, W. T., & Zweig, G. (2013, June). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 746-751).

Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543).

Panigrahi, A., Simhadri, H. V., & Bhattacharyya, C. (2019, July). Word2Sense: sparse interpretable word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 5692-5705).

Park, S., Bak, J., & Oh, A. (2017, September). Rotated word vector representations and their interpretability. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 401-411).

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

Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, *18*(11), 613-620.

Subramanian, A., Pruthi, D., Jhamtani, H., Berg-Kirkpatrick, T., & Hovy, E. (2018, April). Spine: Sparse interpretable neural embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Sun, F., Guo, J., Lan, Y., Xu, J., & Cheng, X. (2016, July). Sparse word embeddings using l1 regularized online learning. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence* (pp. 2915-2921).

Şenel, L. K., Utlu, I., Yücesoy, V., Koc, A., & Cukur, T. (2018). Semantic structure and interpretability of word embeddings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *26*(10), 1769-1779.

Turney, P. D., & Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, *37*, 141-188.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111-3119).

Trifonov, V., Ganea, O. E., Potapenko, A., & Hofmann, T. (2018). Learning and evaluating sparse interpretable sentence embeddings. *arXiv preprint arXiv:1809.08621*.

Wang, Y. X., & Zhang, Y. J. (2012). Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on knowledge and data engineering*, *25*(6), 1336-1353.

Yogatama, M. F. Y. T. D., & Smith, C. D. N. A. (2015). Sparse overcomplete word vector representations. In *ACL*.

Yang, K., & Shahabi, C. (2004, November). A PCA-based similarity measure for multivariate time series. In *Proceedings of the 2nd ACM international workshop on Multimedia databases* (pp. 65-74).