Journal of Biomedical Informatics 77 (2018) 111–119

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Vector representations of multi-word terms for semantic relatedness

Sam Henry*, Clint Cuffy, Bridget T. McInnes

Department of Computer Science, Virginia Commonwealth University, 401 S. Main St., Richmond, VA 23284, USA

ARTICLE INFO

Keywords:
Natural language processing
Semantic similarity and relatedness
Distributional similarity

2010 MSC:
00-01
99-00

ABSTRACT

This paper presents a comparison between several multi-word term aggregation methods of distributional context vectors applied to the task of semantic similarity and relatedness in the biomedical domain. We compare the multi-word term aggregation methods of summation of component word vectors, mean of component word vectors, direct construction of compound term vectors using the compoundify tool, and direct construction of concept vectors using the MetaMap tool. Dimensionality reduction is critical when constructing high quality distributional context vectors, so these baseline co-occurrence vectors are compared against dimensionality reduced vectors created using singular value decomposition (SVD), and word2vec word embeddings using continuous bag of words (CBOW), and skip-gram models. We also find optimal vector dimensionalities for the vectors produced by these techniques. Our results show that none of the tested multi-word term aggregation methods is statistically significantly better than any other. This allows flexibility when choosing a multi-word term aggregation method, and means expensive corpora preprocessing may be avoided. Results are shown with several standard evaluation datasets, and state of the results are achieved.

1. Introduction

Semantic similarity and relatedness measures quantify the degree to which two concepts are similar (e.g., *liver-organ*) or related (e.g., *headache-aspirin*). These measures are critical to improving many tasks, such as the retrieval [1] and clustering [2] of biomedical and clinical documents, and the development of biomedical terminologies and ontologies [3]. There are many ways to quantify the relatedness or similarity between terms [4], one such method is to compute the similarity between two terms' distributional context vectors [5]. A word's distributional context vector is constructed from its co-occurrences between surrounding words in a corpus, its contexts. The assumption is that related words will have similar contexts. Distributional context vectors have been shown to perform well on semantic similarity and relatedness tasks [6–8], but since these vectors are built for individual words, the best method to represent multi-word terms (e.g. New York City, or Heart Attack) is unclear. To answer this question, this paper explores several aggregation methods for vector representations of multi-word terms.

An answer to this question would be incomplete without addressing the challenges of sparseness and noise of distributional context vectors. We define sparseness as the vectors contain mostly zero values, and noise as information that is overly general and does not contribute to the word's representation. Dimensionality reduction techniques may be applied to transform the data from a higher dimensional space to a lower dimensional space, in which sparseness and noise are reduced. Most previous methods exploring feature extraction for quantifying the relatedness between biomedical and clinical term pairs have ignored multi-word terms [9–11]. We analyze several dimensionality reduction techniques for each multi-word term aggregation method, and vary the vectors dimensionality parameter for these techniques. Specifically, the contributions of this paper are an analysis of:

- Multi-Word Term Aggregation Methods: We compare the performance of summation of component word vectors, averaging of component word vectors, direct creation of multi-word term vectors using the compoundify tool, and direct creation of concept vectors using the MetaMap tool.
- Dimensionality Reduction Techniques: singular value decomposition (SVD), word embeddings using skip-gram, and word embeddings using continuous bag of words (CBOW) are evaluated as dimensionality reduction techniques. Explicit vectors of word-to-word, term-to-term, or concept-to-concept co-occurrences are used as a baseline.
- Vector Dimensionality: the dimensionality of the generated vectors is a parameter that effects performance. We evaluate each multi-word term aggregation method's and dimensionality reduction technique's performance at dimensionalities of 100, 200, 500, 1000, and 1500. For SVD we also evaluate at dimensionalities of 2000, 2500, and 3000.

* Corresponding author.
E-mail address: henryst@vcu.edu (S. Henry).

https://doi.org/10.1016/j.jbi.2017.12.006
Received 25 July 2017; Received in revised form 9 October 2017; Accepted 12 December 2017
Available online 13 December 2017
1532-0464/

Vectors are generated from MEDLINE abstracts and titles. We achieve state of the art results on the UMNSRS and MiniMayoSRS evaluation standards, and find that direct creation of concept vectors achieves the highest sum of correlations across all datasets, but only marginally. We find no statistical significance between any multi-word term aggregation method across all dimensionality reduction techniques, and dimensions tested, indicating that choosing a multi-word term aggregation method is arbitrary, and the amount of text preprocessing required may be decided by the needs of other system components. The best results are achieved using concept vectors with the dimensionality reduction method of word2vec word embeddings using CBOW with a dimensionality of 200.

This paper is organized as follows: First an overview of related work is presented, followed by an overview of the tools and datasets used. Next the method is presented in detail. After that a lengthy results and discussion section is presented, in which the dimensionality reduction techniques are compared, optimal vector dimensionalities are found, multi-word term aggregation methods are compared, and lastly our results are compared to other authors' who have used similar techniques. This paper ends with our conclusions and proposals for future work.

2. Background

2.1. Related work

Most work with relatedness measures in the biomedical domain is based on distributional statistics. For this, a context vector is created for each concept and the similarity between the concepts is calculated by taking the cosine between their individual context vectors. Patwardhan and Pedersen [12] use second-order co-occurrence vectors first introduced by Schutze [13] to deal with sparseness of the explicit co-occurrence vectors. In this method, a vector containing co-occurrences within a corpus is created for each word in a word's definition. These word vectors are averaged to create a single co-occurrence vector for the concept. Liu et al. [14] modify and extend this measure to quantify the relatedness between biomedical and clinical terms in the UMLS.

Recently, word embeddings (such as word2vec) have become an increasingly popular vector representation technique [15,9–11]. Although previous work explores different parameters and corpora, multi-word term aggregation methods have been ignored, and multi-word terms are often dropped from the evaluation standards to account for the inability to represent such terms [9–11]. Table 1 summarizes the contributions of previous authors' work with word embeddings for semantic similarity and relatedness.

2.2. Vector representations and dimensionality reduction techniques

2.2.1. Explicit co-occurrence vectors

Distributional context vectors model term meaning based on the context in which they are seen. The simplest method, explicit co-occurrence vectors create vector representations that have dimensionality the size of the vocabulary. These vectors are constructed over a training corpus and record all co-occurrences of a word within a predefined window size. For example, consider the previous sentence; the vector for the word "record" would increment values at the vector indexes for the terms "all" with a window size of 1, "all" and "co-occurrences" with a window size of 2, and "all", "co-occurrences", and "of" with a window size of 3. These vectors are prone to sparseness and noise, which obscure term meaning. Dimensionality reduction techniques may be applied to reduce sparseness and noise, thereby better modeling word meaning. When applying a dimensionality reduction technique, the dimensionality of the resulting vector is a parameter. A dimensionality that is too small will not be able to effectively differentiate between words in vector space. A dimensionality that is too large does not successfully solve the problems of sparsity and noise. We aim to find the optimal vector dimensionality, that is, the most compact vector representation with the best performance.

Table 1
Summary of related work using word embeddings for semantic relatedness in the biomedical domain. The citation column indicates the author and reference. The method column indicates whether the author used CBOW, skip-gram (SG), or both for their evaluation. The training dataset(s) column shows the training corpora used in the experiments, and the hyperparams column indicates whether the author reported results with various hyperparameter settings (e.g. vector dimensionality, window size, etc.).

	Method			
Citation	CBOW	SG	Training Dataset(s)	Hyperparams
Sajadi et al. [15]		x	OHSUMED	
Muneeb et al. [9]	x	x	PMC	x
Chiu et al. [10]	x	x	Pubmed, PMC	x
Pakhomov et al. [11]	x		Fairview, PMC, Wiki	

2.2.2. Singular value decomposition

We construct a co-occurrence matrix, M using the explicit vectors of each word in a corpus as a row in the matrix. Deerwester et al. [16] proposed Latent Semantic Indexing (LSI) which reduces dimensionality using the factor analysis technique, singular value decomposition (SVD). SVD decomposes a matrix, M into a product of three simpler matrices, such that $M = U{\cdot}\Sigma{\cdot}V^T$. The matrices U and V are orthonormal and Σ is a diagonal matrix of eigenvalues in decreasing order. Limiting the eigenvalues to d, we can reduce the dimensionality of our matrix to $M_d = U_d{\cdot}\Sigma_d{\cdot}V_d^T$. The columns of U_d correspond to the eigenvectors of M_d. Typically this decomposition is achieved without any loss of information. Here though, SVD reduces a word-by-word co-occurrence matrix from thousands of dimensions to hundreds, and therefore the original matrix cannot be perfectly reconstructed from the three decomposed matrices. The intuition is that any information lost is noise, the removal of which causes the similarity and non-similarity between words to be more discernible [17].

2.2.3. Word embeddings

Word embeddings are another, increasingly popular [18] dimensionality reduction method. Word embeddings construct reduced dimensionality distributional context vectors directly from a training corpus by iterating over it and learning word representations. The word embeddings method, *word2vec*, proposed by Mikolov et al. [19], is a neural network based approach that learns a series of weights (the hidden layer within the neural network) that either maximizes the probability of a word given the surrounding context, referred to as the continuous bag of words (CBOW) approach, or to maximize the probability of the context given a word, referred to as the skip-gram approach. For either approach, the resulting hidden layer consists of a matrix where each row represents a word in the vocabulary and columns a word embedding. The basic intuition behind this method is that words closer in meaning will have vectors closer to each other in this reduced space.

3. Tools

3.1. Unified medical language system

The Unified Medical Language System (UMLS) is a data warehouse containing three knowledge sources: the Metathesaurus, the Semantic Network and the SPECIALIST Lexicon. The Metathesaurus contains approximately 3.1 million biomedical and clinical concepts from over 100 different terminologies that have been semi-automatically integrated into a single source. Each concept may have multiple terms that map to it, and is uniquely identified with an assigned Concept Unique Identifier (CUI). The SPECIALIST Lexicon provides lexical information of commonly used English words and biomedical vocabulary,

about half of which are multi-word terms (472,608 single word terms and 462,668 multi-word terms). Each entry term includes syntactic, morphological, and orthographic information, including spelling variants.

3.2. MetaMap

MetaMap [20] is a tool developed by the National Library of Medicine (NLM), National Institutes of Health (NIH) to map raw text to UMLS concepts (CUIs). From the raw text input, it produces a set of ordered CUI mappings. This has the effect of performing stop word removal and text normalization. MetaMap is freely available and can be used through an interactive web interface, or downloaded for local use. [1] MetaMap can be run on any text, but since MEDLINE is a popular dataset, MetaMap is run on MEDLINE to produce the MetaMapped MEDLINE baseline, [2] which contains the text of all titles and abstracts of MEDLINE mapped to UMLS concepts.

3.3. Text::NSP

We create co-occurrence matrices using the Text::NSP packaged developed by Pedersen et al. [21]. Text::NSP is a freely available open-source software package that identifies n-grams, collocations, and word associations in text. It is implemented in Perl and takes advantage of regular expressions to provide very flexible tokenization.

3.4. Compoundify

Compoundify combines multi-word terms in text using the UMLS SPECIALIST Lexicon as a glossary. The result is a corpus in which all compound terms have been identified (component words joined by an underscore). Multi-word term vectors are then directly constructed from this text. Compoundify is a part of the word2vec-interface package, [3] a Perl interface to word2vec [19]. Word2vec-interface version 0.03 is used.

4. Data

4.1. Training data

We develop our word, term, and concept vectors using co-occurrence information from titles and abstracts of the 2015 MEDLINE baseline. [4] MEDLINE is a bibliographic database containing over 23 million citations to journal articles in the biomedical domain and is maintained by National Library of Medicine. The 2015 MEDLINE Baseline encompasses approximately 5600 journals, and contains 22,775,609 citations, of which 13,835,206 contain abstracts. In this work, we use MEDLINE abstracts and titles from 1975 to present day. Prior to 1975, only 2% of the citations contained an abstract.

4.2. Evaluation data

We evaluate our method on two reference standards [5]: the UMNSRS tagged for similarity (UMNSRS Sim.), the UMNSRS tagged for relatedness (UMNSRS Rel.), and the MiniMayoSRS tagged for relatedness by physicians (MiniMayo Phy.) and by medical coders (MiniMayo Cod.). These datasets are sets of term pairs and a human assigned score. Each term pair may contain single-word or multi-word terms, and the assigned score is the average between all assessers. For example, the MiniMayoSRS tagged for similarity contains the following *score<>term<>term* triplets: “3.3<>Heart<>Myocardium”, “2.7<>Tumor metastasis<>Adenocarcinoma”, “2.0<>Brain tumor<>Intracranial hemorrhage”.

MiniMayoSRS: MayoSRS, developed by Pakhomov et al. [22], consists of 101 clinical term pairs whose relatedness was determined by nine medical coders and three physicians from the Mayo Clinic. The relatedness of each term pair was assessed based on a four point scale: (4.0) practically synonymous, (3.0) related, (2.0) marginally related and (1.0) unrelated. MiniMayoSRS is a subset of the MayoSRS and consists of thirty term pairs on which a higher inter-annotator agreement was achieved. The average correlation between physicians is 0.68. The average correlation between medical coders is 0.78. Twenty of the thirty term pairs (66.67%) contain a multi-word term. We evaluate our method on the mean of the physician scores, and the mean of the coders scores in this subset in the same manner as reported by Pedersen et al. [4].

UMNSRS: UMNSRS, developed by Pakhomov et al. [23], consists of 725 clinical term pairs whose semantic similarity and relatedness was determined independently by four medical residents from the University of Minnesota Medical School. The similarity and relatedness of each term pair was annotated based on a continuous scale by having the resident touch a bar on a touch sensitive computer screen to indicate the degree of similarity or relatedness. The Intraclass Correlation Coefficient (ICC) for the reference standard tagged for similarity was 0.47, and 0.50 for relatedness. Therefore, as suggested by Pakhomov and colleagues, we use a subset of the ratings consisting of 401 pairs for the similarity set and 430 pairs for the relatedness set which each have an ICC of 0.73. Twenty (4.99%) and seventeen (3.95%) of the term pairs contain multi-word terms for the similarity and relatedness subsets respectively.

5. Method

Fig. 1 shows the overall method used to generate relatedness scores. The major steps in this process are: (1) preprocess the text, (2) create vector representations, (3) aggregate terms, and (4) calculate relatedness. The arrows show how vector representations flow through the process, and the boxes show different steps in the process. In this section, we discuss each major step.

5.1. Corpus preprocessing

We use the 2015 MEDLINE baseline as the training corpus for all techniques, however different multi-word term aggregation methods, and different dimensionality reduction techniques require different preprocessing steps. Fig. 1 shows how each vector representation was obtained. First, the raw 2015 MEDLINE corpus, and the 2015 MetaMapped MEDLINE corpus are downloaded for the years 1975 onward. The 2015 MetaMapped MEDLINE baseline is used directly, without further preprocessing to generate CUI term aggregation vectors, but the raw (not MetaMapped) 2015 MEDLINE corpus must be preprocessed for the other methods. The raw corpus is converted to all lowercase, and all apostrophe s’s (‘s) are removed. This normalized corpus is used as input for each dimensionality reduction technique to generate the word vectors used in the sum and mean aggregation methods. To generate compound vectors, this normalized corpus is compoundified using the compoundify tool.

5.2. Vector representation and dimensionality reduction

We used the following packages and settings to obtain our vector representations and perform dimensionality reduction:

[1] Explicit Representation: We used the Text::NSP package to create a co-occurrence matrix, the rows of which are **explicit** vector

[1] https://metamap.nlm.nih.gov/MetaMap.shtml.
[2] https://mbr.nlm.nih.gov/Download/MetaMapped_Medline/2015/.
[3] https://sourceforge.net/projects/word2vec-interface/.
[4] http://mbr.nlm.nih.gov/Download/index.shtml.
[5] http://rxinformatics.umn.edu/SemanticRelatednessResources.html.

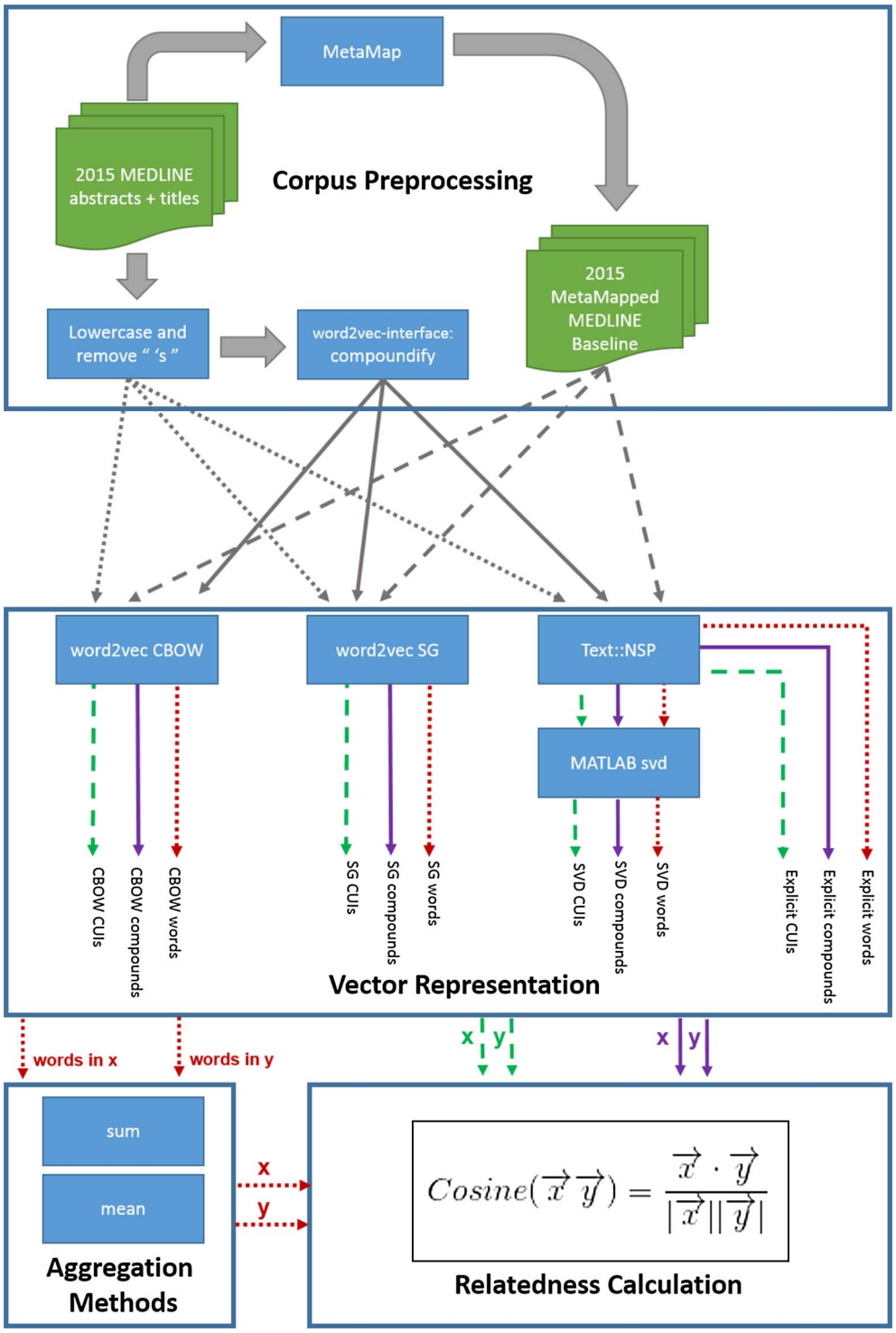

Fig. 1. Procedure for generating vector representations using different dimensionality reduction techniques and term aggregation methods.

representations for terms. We used a windows size of 8, a frequency cutoff of 5, and removed stopwords (stopword removal is a package option, for a complete list of stop words see the Text::NSP package).

[2] Singular Value Decomposition. We ran the MATLAB R2016b implementation of sparse matrix **SVD** (svds) on the explicit representation matrix, and used each row of the resulting *U* matrix as a reduced vector.

[3] Word Embeddings: We used the *word2vec* package developed by Mikolov et al. [19] for the continuous bag of words (**CBOW**) and skip-gram word (**SG**) embedding models with a window size of 8, a frequency cutoff of 5, and default settings for all other parameters.

5.3. Aggregation methods

To compute the relatedness of multi-word terms using distributional context vectors, an aggregation method must be used. Typically, distributional context vectors are created for individual words in a corpus. For example, the words "heart" and "attack" are represented by their vector representations, $\overrightarrow{heart}$ and $\overrightarrow{attack}$. These component word vectors of a multi-word term must be combined to create a single vector for that term ($\overrightarrow{heart attack}$). We compare two combine operations:

1. **sum** – the multi-word term vector is the summation of the

component word vectors - $\overrightarrow{heart} + \overrightarrow{attack} = \overrightarrow{heart\_attack}$

2. **mean** – the multi-word term vector is the average of the component word vectors - $(\overrightarrow{heart} + \overrightarrow{attack})/2 = \overrightarrow{heart\_attack}$
   Rather than combine word vectors after construction, multi-word term vectors may be constructed directly from a preprocessed training corpus in which multi-word terms have been identified. For example, in the training corpus, any occurrence of "heart" followed by "attack" will be tagged as the multi-word term "heart_attack" rather than either of the component word vectors. A heart_attack vector may then be directly from this corpus. We compare two preprocessing methods for direct construction of multi-word term vectors:
3. **compounds** – text is preprocessed using the compoundify tool. Multi-word term vectors are directly constructed from this compoundified text.
4. **CUIs** – the MetaMapped MEDLINE baseline is used, which contains text that has been preprocessed using the MetaMap tool. MetaMap maps raw text to UMLS concepts (CUIs) resulting in an ordered list of CUIs, from which multi-word concept vectors are directly constructed. A complication of MetaMap is that multi-word terms may be ambiguous. In these cases, MetaMap will generate all possible concepts that map to that term. When this occurs, we replace the term with each possible concept mapping. For example, the phrase "penicillin tolerant E-Coli" will be replaced with "C0220892 C0013220 C0020963 C0014834", where "penicillin" is replaced C0220892, the ambiguous term, "tolerant" is replaced with both concept mappings, C0013220 and C0020963, corresponding to drug tolerance and immune tolerance respectively, and "E-Coli" is replaced with C0014834.

5.4. Relatedness, correlation, and significance calculations

We use cosine distance between the distributional context vectors of each term to quantify the relatedness of the term pair. The correlations between the generated relatedness scores and the human-assigned scores are calculated using Spearman's Rank Correlation (ρ). Spearman's measures the statistical dependence between two variables to assess how well the relationship between the rankings of the variables can be described using a monotonic function. We used Fisher's R-to-Z transformation [24] to calculate the significance between the correlation results. We use the Word2vec::Interface package [6] version 0.03 to obtain the disambiguation accuracy for each of the WSD datasets. The differences between the means of disambiguation accuracy were tested for statistical significance using pair-wise Student's t-test.

6. Results and discussion

In this section, we discuss the results of the various experiments. We compare dimensionality reduction techniques (SG, CBOW, SVD, explicit), compare different dimensionalities of the reduced vectors (100, 200, 500, 1000, 1500), and compare multi-word term aggregation methods (sum, mean, compounds, CUIs). We also compare our results to similar previous works. Statistical significances are used throughout this section; here we define significant, or significance, to mean statistically significantly different correlations with $p < 0.05$. Table 2 shows all of the results for each term aggregation method, dimensionality reduction method, and vector dimensionality tested. Values in each cell show the correlation, slash, n, the number of samples compared. A hyphen ('-') indicates a score could not be calculated using those parameters. CBOW at a dimensionality of 1500 could not be calculated due to errors in the word2vec package. The first column ("100/e") shows results for a dimensionality of 100, and results with explicit vector representation (for which dimensionality is the vocabulary size and does not vary). A discussion and analysis of these results is presented in the next few subsections.

6.1. Comparison between dimensionality reduction techniques

Here, we compare the different dimensionality reduction techniques of SVD, SG, and CBOW against each other, and against the baseline of explicit. Fig. 2 shows the highest correlation of each of these techniques, for each dataset. Here, the highest correlation indicates the highest correlation among any results generated for that technique regardless of dimensionality or aggregation method. Table 3 shows statistical significances among the scores shown in Fig. 2.

CBOW generates the highest overall accuracy, with a sum of correlations at 3.03. Although CBOW's overall accuracy is slightly higher than SG, SG outperforms CBOW on some datasets, and CBOW and SG do not perform significantly different on any single dataset. CBOW is the only method to perform significantly better than explicit on MiniMayo Phys., or MiniMayo Cod. The word embeddings methods SG and CBOW, generate statistically significant higher correlations than SVD and explicit on UMNSRS Sim. and UMNSRS Rel. SVD only generates a significantly higher correlation than explicit for a single dataset (UMNSRS Sim.). **Conclusion:** all the dimensionality reduction techniques improve correlation accuracy, but the word embeddings approaches (SG and CBOW) perform significantly better than SVD and explicit. SG and CBOW perform on par with each other, with no significant differences. CBOW does, however generate the vector representations much more quickly than SG (our rough estimates indicate that SG takes between 5 and 9 times as long to train), and may be preferred due to this decreased computation time.

6.2. Comparison between vector dimensionality

Here, we tested the effects of vector dimensionality on performance for each dimensionality reduction technique. Beginning with a vector dimensionality of 100, we increased the dimensionality of each vector representation to values of 200, 500, 1000, and 1500. Due to errors generated by the word2vec package, we were unable to generate vectors with CBOW at a dimensionality of 1500, although to more comprehensively test SVD, we continued to increase the dimensionality of SVD vectors to 2000, 2500, and 3000. Fig. 3shows the correlations of each technique on each dataset as the dimensionality increases. Each sub-figure shows performance on a different dataset. We show the highest correlation for each dimensionality reduction method and vector dimensionality, regardless of aggregation method. These results show that SG's and CBOW's correlations remain nearly constant as dimensionality increases, indicating that vector dimensionality has little impact on their performance, and a dimensionality of 200 is sufficient for these methods. For SVD, we do see an increase in performance as dimensionality increases. The performance continues to increase to 1500, and to determine whether a dimensionality greater than 1500 would continue to increase performance, we created additional SVD vectors with dimensionalities of 2000, 2500, and 3000. Fig. 4 shows the overall performance of SVD at each vector dimensionality. The values above each column indicates the sum of the dataset correlations. We see an increase in correlation up to a dimensionality 1000, at which point correlation scores remain constant at 1500, and decrease at 2000, 2500, and 3000, indicating a dimensionality of 1000 is sufficient for SVD. **Conclusion:** a dimensionality of 200 is sufficient for the word embeddings methods, SG and CBOW. A larger dimensionality of 1000 is sufficient for SVD.

6.3. Comparison between term aggregation methods

Here, we compare the multi-word term aggregation methods, sum, mean, compounds, and CUIs. Fig. 5 shows the highest correlations of different aggregation methods across all dimensionality reduction

[6] http://search.cpan.org/dist/Word2vec-Interface/.

Table 2
Results of each term aggregation method, dimensionality reduction technique, and vector dimensionality on all datasets. Values in each cell show the correlation, slash, *n*, the number of samples compared. A hyphen ('-') indicates a score could not be calculated using those parameters. The first column ("100/e") shows results for a vector dimensionality of 100, and results with explicit vector representation. Bolded scores indicate the highest performing combination of parameters in each box.

		MiniMayo Phys. Dimensionality					**MiniMayo Cod. Dimensionality**				
Aggreg.	**Red.**	100/e	200	500	1000	1500	100/e	200	500	1000	1500
Sum	SG	0.78/29	0.79/29	0.74/29	0.76/29	0.74/29	0.79/29	0.80/29	0.78/29	0.79/29	0.78/29
	CBOW	0.81/29	**0.82**/29	0.79/29	0.75/29	-	**0.82**/29	**0.82**/29	0.79/29	0.78/29	-
	SVD	0.38/28	0.57/28	0.56/28	0.79/28	0.66/28	0.36/28	0.53/28	0.52/28	0.54/28	0.71/28
	Explicit	0.37/28	-	-	-	-	0.34/28	-	-	-	-
Mean	SG	0.78/29	0.79/29	0.74/29	0.76/29	0.74/29	0.79/29	0.80/29	0.78/29	0.79/29	0.78/28
	CBOW	0.81/29	**0.82**/29	0.79/29	0.75/29	-	**0.82**/29	0.81/29	0.79/29	0.78/29	-
	SVD	0.37/29	0.52/29	0.54/29	0.77/29	0.65/29	0.36/29	0.53/29	0.53/29	0.54/29	0.71/29
	Explicit	0.34/28	-	-	-	-	0.36/29	-	-	-	-
Compound	SG	0.78/28	0.78/28	0.77/28	0.76/28	0.75/28	0.75/28	0.76/28	0.76/28	0.75/28	0.76/28
	CBOW	0.79/28	**0.80**/28	0.79/28	0.77/28	-	0.76/28	**0.78**/28	**0.78**/28	**0.78**/28	-
	SVD	0.65/28	0.74/28	0.75/28	0.72/28	0.70/28	0.65/28	0.73/28	0.70/28	0.72/28	0.72/28
	Explicit	0.49/28	-	-	-	-	0.51/28	-	-	-	-
Cui	SG	0.76/29	0.76/29	0.77/29	0.76/29	0.76/29	0.77/29	0.77/29	0.78/29	0.77/29	0.79/29
	CBOW	0.77/29	0.75/29	**0.78**/29	0.76/29	-	**0.83**/29	**0.83**/29	**0.83**/29	0.82/29	-
	SVD	0.41/28	0.42/28	0.50/28	0.40/28	0.38/28	0.35/28	0.39/28	0.58/28	0.48/28	0.35/28
	Explicit	0.37/28	-	-	-	-	0.26/28	-	-	-	-
		UMNSRS Rel.					**UMNSRS Sim.**				
		100/e	200	500	1000	1500	100/e	200	500	1000	1500
Sum	SG	**0.70**/374	**0.70**/374	0.68/374	0.69/374	0.68/374	0.59/396	0.61/396	**0.62**/396	**0.62**/396	**0.62**/396
	CBOW	0.68/374	0.69/374	0.66/374	0.61/374	-	0.55/396	0.61/396	0.61/396	0.58/396	-
	SVD	0.53/331	0.52/331	0.55/331	0.56/331	0.52/331	0.41/343	0.36/343	0.45/343	0.47/343	0.45/343
	Explicit	0.46/331	-	-	-	-	0.42/343	-	-	-	-
Mean	SG	**0.70**/374	**0.70**/374	0.68/374	0.69/374	0.68/374	0.58/397	0.60/397	**0.61**/397	**0.61**/397	**0.61**/397
	CBOW	0.68/374	0.69/374	0.66/374	0.61/374	-	0.55/397	0.59/397	0.59/397	0.57/397	-
	SVD	0.53/332	0.52/332	0.55/332	0.55/32	0.52/332	0.39/346	0.34/346	0.46/346	0.47/346	0.43/346
	Explicit	0.33/400	-	-	-	-	0.36/430	-	-	-	-
Compound	SG	**0.72**/373	0.71/373	0.70/373	0.69/373	0.70/373	0.63/393	0.64/393	0.64/393	0.65/393	**0.66**/393
	CBOW	0.70/373	0.70/373	0.68/373	0.65/373	-	0.62/393	0.64/393	0.65/393	0.65/393	-
	SVD	0.49/328	0.51/328	0.58/328	0.60/328	0.58/328	0.39/335	0.38/335	0.48/335	0.54/335	0.52/335
	Explicit	0.45/328	-	-	-	-	0.45/335	-	-	-	-
Cui	SG	**0.74**/388	**0.74**/388	**0.74**/388	**0.74**/388	**0.74**/388	0.62/413	0.62/413	0.63/413	**0.64**/413	**0.64**/413
	CBOW	0.72/388	0.73/388	0.73/388	0.72/388	-	0.56/413	0.56/413	0.59/413	0.60/413	-
	SVD	0.41/362	0.45/362	0.50/362	0.53/362	0.57/362	0.26/380	0.31/380	0.30/380	0.34/380	0.38/380
	Explicit	0.35/362	-	-	-	-	0.20/380	-	-	-	-

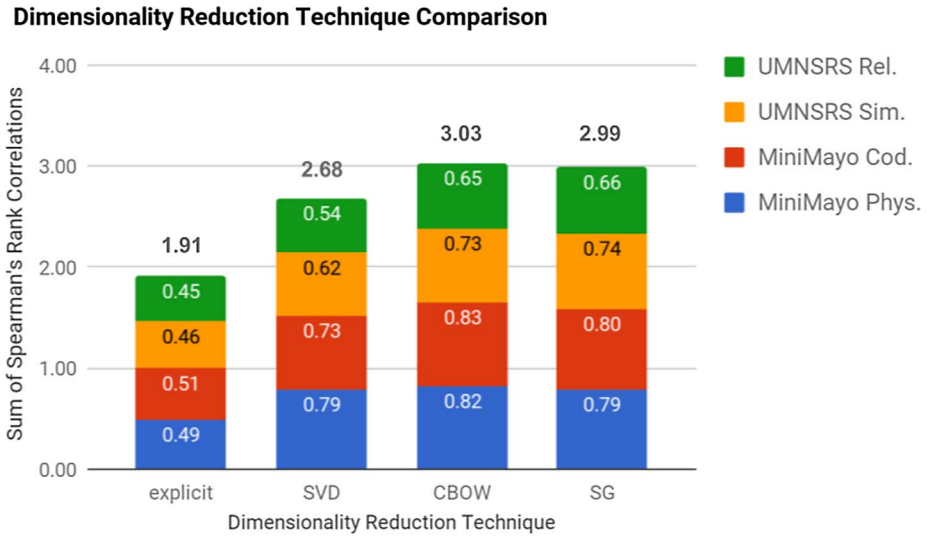

Fig. 2. Best results for each dimensionality reduction technique. The correlation for each dataset is shown within its rectangle, and the sum of correlations is shown above each column. A sum of 4.0 would indicate a perfect correlation of 1.0 for every dataset.

Table 3
The two tailed p-values using Fishers R-to-Z transform, comparing the results of each dimensionality reduction technique. Each table corresponds to a different dataset, each row and column a different dimensionality reduction technique. p-values less than 0.05 are marked with an asterisk (*).

	Explicit	SVD	SG		Explicit	SVD	SG
	MiniMayo Phys.				**MiniMayo Cod.**		
SVD	0.0588	-	-	SVD	0.1971	-	-
SG	0.0561	1.0	-	SG	0.0561	0.5419	-
CBOW	0.0264*	0.7642	0.7566	CBOW	0.0257*	0.3524	0.749
	UMNSRS Sim.				**UMNSRS Rel.**		
SVD	0.0029*	-	-	SVD	0.1236	-	-
SG	0.0*	0.0021*	-	SG	0.0*	0.0114*	-
CBOW	0.0*	0.0054*	0.7642	CBOW	0.0001*	0.022*	0.8103

techniques, and dimensionalities for each dataset. From this, we see that CUIs achieves the highest sum of correlations, but only by a small amount, and is not the best performing method on all datasets. Each method performs the best (or ties) on different datasets: sum and mean tied for highest on the MiniMayo Phys. dataset, compounds on the UMNSRS Rel. dataset, and CUIs on the UMNSRS Sim. and MiniMayo Cod. datasets. Table 4 shows the p-values for the correlations shown in Fig. 5. No aggregation method performs significantly better on any single dataset, and the sum of correlations for CUIs is only marginally higher than other methods. **Conclusion:** there is no significantly best aggregation method, when comparing over all parameters.

To further analyze the aggregation methods, we set the dimensionality reduction technique, and vector dimensionality parameters to our recommendations found in previous subsections. That is, we compared results of each multi-word term aggregation method using dimensionality reduction of word embeddings using CBOW and a vector dimensionality of 200. Table 5 (restated from Table 2) shows these results. Table 6 shows the statistical significance between these results. No multi-word term aggregation method achieves the highest correlation on all datasets, and each method achieves (or ties) the highest result for one of the datasets. There is no statistical significance between any of the techniques on any of the datasets. After preprocessing, the computational cost of each method is nearly identical. Ambiguity is only an upfront problem for concept vectors, but they also have the advantage of mapping synonymous terms to the same concept before vector construction. This is likely why concept vectors (CUI) achieve the highest correlation on two datasets (MiniMayo Cod., and UMNSRS Rel.), and is why they are able to compare more terms (higher *n*). Concept mapping is however, an expensive preprocessing step, and it is unclear how to best handle ambiguous concept mappings. Using compounds requires less expensive data preprocessing, and since it does not map synonymous terms to the same concept, the problem of ambiguity is essentially ignored. Using sum or mean requires the least preprocessing, and does not rely on external lexicons to construct mutliword term vectors, ambiguity is ignored, and vectors can be created for any term in any domain. **Conclusion:** There is no significantly best multi-word term aggregation method across either all parameters or recommended parameters. Using concept vectors achieves the highest correlation on more datasets and it is able to compare more terms. It is therefore slightly favored, but requires the term to map to a concept. Sum or mean provide more flexibility and less preprocessing.

6.4. Comparison with previous work

Here, we summarize and compare previous works that use word2vec word embeddings for semantic similarity and relatedness tasks. These results are not directly comparable to each other, or ours due to different term pairs used. Table 7 summarizes both their results and ours. Sajadi et al. [15] trained the word2vec skip-gram model over UMLS Concepts (CUIs) identified by MetaMap on the OHSUMED corpus, a collection of 348,566 biomedical research articles. They evaluated the algorithm on the UMNSRS reference standard, Minimayo

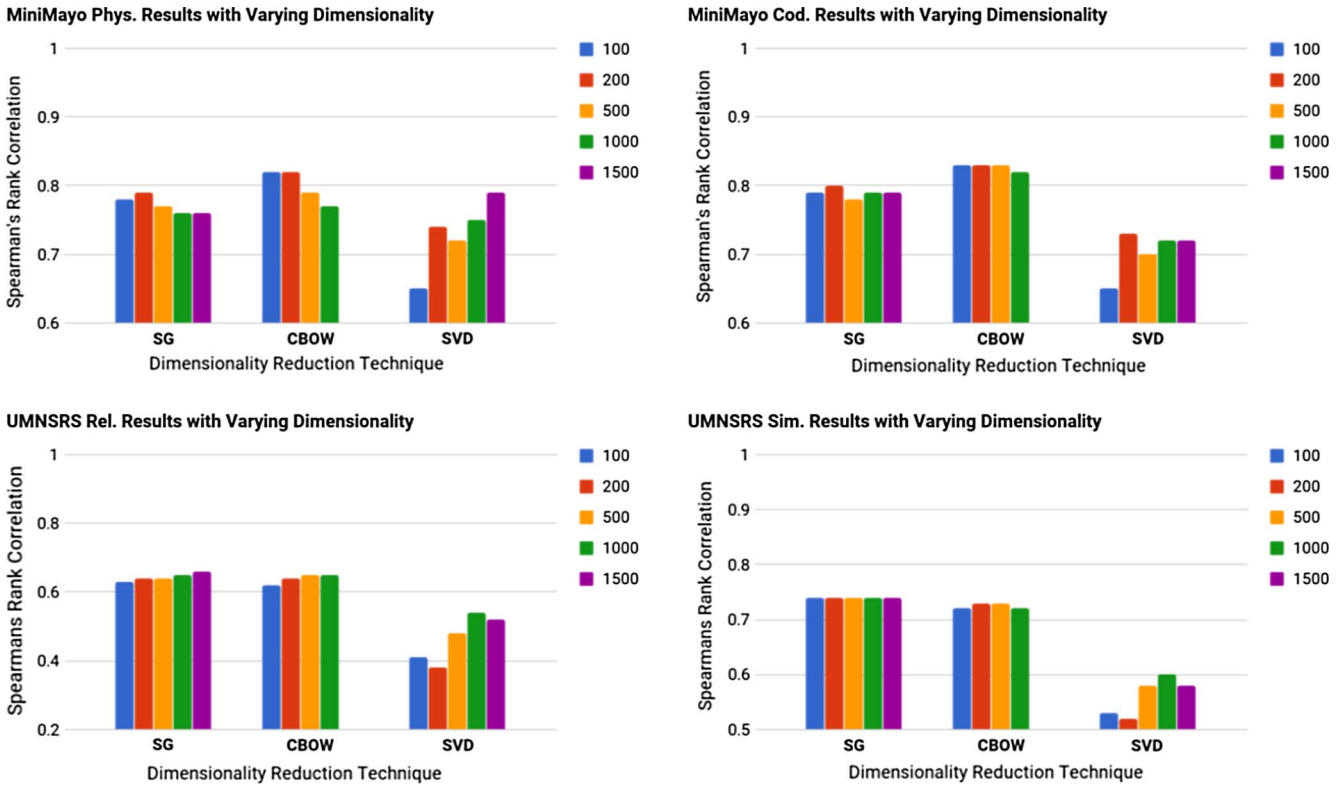

Fig. 3. Results of varying vector dimensionality on each datasets. The subgraphs correspond to a single dataset, and the column groupings a dimensionality reduction technique. Different colored columns indicate a different vector dimensionality. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

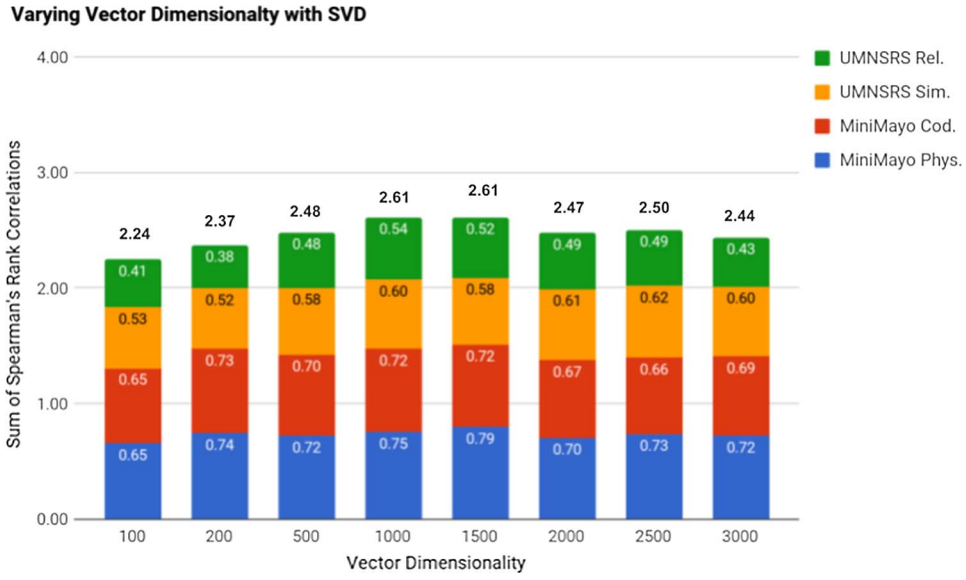

Fig. 4. Sum of results across datasets for each vector dimensionality tested with SVD. Each column corresponds to a vector dimensionality, individual dataset correlations are shown within the colored rectangles, and sum of correlations are shown above each column. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

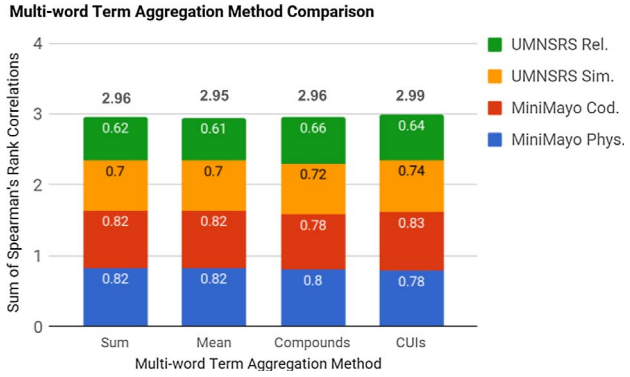

Fig. 5. Results of different aggregation methods on each dataset.

Table 4
The two tailed p-values using Fishers R-to-Z transform of each term aggregation method's correlation scores of each dataset. Each table corresponds to a different dataset, each row and column a different term aggregation method.

	Sum	Mean	Compound		Sum	Mean	Compound
	MiniMayo Phys.				**MiniMayo Cod.**		
Mean	1.0	-	-	Mean	1.0	-	-
Comp.	0.8337	0.8337	-	Comp.	0.5823	0.5823	-
CUI	0.6892	0.6892	0.8493	CUI	0.2543	0.2543	0.5552
	UMNSRS Sim.				**UMNSRS Rel.**		
Mean	1.0	-	-	Mean	0.8181	-	-
Comp.	0.5823	0.5823	-	Comp.	0.3421	0.242	-
CUI	0.2543	0.2543	0.5552	CUI	0.6384	0.4839	0.6241

Table 5
Results of each multi-word term aggregation technique using the recommended settings of word embeddings using CBOW at a vector dimensionality of 200.

	Sum	Mean	Compound	CUI
	CBOW Results at Vector Dimensionality of 200			
MiniMayo Cod.	0.82/29	0.81/29	0.78/28	0.83/29
MiniMayo Phys.	0.82/29	0.82/29	0.80/28	0.75/29
UMNSRS Sim.	0.61/396	0.59/397	0.64/393	0.56/413
UMNSRS Rel.	0.69/374	0.69/374	0.70/373	0.73/388

Table 6
The two tailed p-values using Fishers R-to-Z transform of the correlation scores of multi-word term aggregation methods using recommended parameters. Each table corresponds to a different dataset, each row and column a different term aggregation method using CBOW and a dimensionality of 200.

	Sum	Mean	Compound		Sum	Mean	Compound
	MiniMayo Phys.				**MiniMayo Cod.**		
Mean	1.0	-	-	Mean	0.9124	-	-
Comp.	0.4168	0.4168	-	Comp.	0.6892	0.3859	-
CUI	0.5093	0.5093	0.8337	CUI	0.9124	0.8259	0.6101
	UMNSRS Sim.				**UMNSRS Rel.**		
Mean	0.6599	-	-	Mean	1.0	-	-
Comp.	0.4902	0.2585	-	Comp.	0.7949	0.7949	-
CUI	0.2801	0.5222	0.0767	CUI	0.2670	0.2670	0.4009

Table 7
Comparison with Previous Work. Previous authors' results using word2vec word embeddings, compared to our results. "-" indicate that results were not reported for that dataset. Chiu et al. do not report n. Each column corresponds to a different dataset. The "avg" column reports the average of physician's and coder's scores for the MiniMayoSRS dataset.

Method	UMNSRS		MiniMayoSRS		
	Sim.	Rel.	Phys.	Cod.	Avg.
CBOW words (ours)	0.61 (n = 396)	0.69 (n = 374)	0.82 (n = 29)	0.82 (n = 29)	0.82
CBOW comp. (ours)	0.65 (n = 393)	0.70 (n = 373)	0.80 (n = 29)	0.78 (n = 28)	0.79
CBOW cuis (ours)	0.60 (n = 413)	0.73 (n = 388)	0.77 (n = 29)	0.83 (n = 29)	0.80
Sajadi et al. [15]	0.39 (n = 566)	0.39 (n = 597)	-	-	0.8
Pakhomov et al. [11]	0.62 (n = 449)	0.58 (n = 458)	-	-	-
Muneeb et al. [9]	0.52 (n = 462)	0.45 (n = 465)	-	-	-
Chiu et al. [10]	0.65 (n = ?)	0.60 (n = ?)	-	-	-

reference standard, and Mayo reference standard.

Muneeb et al. [9] trained both the skip-gram and continuous bag of words (CBOW) word2vec models over the PubMed Central Open Access (PMC) corpus of approximately 1.25 million articles using a window size of 9. They evaluated the models on a subset of the UMNSRS data,

removing word pairs that did not occur in the corpus more than ten times. The authors evaluated vector dimensionalities of 25, 50, 100, and 200, finding that a dimensionality of 200 performed better than the lower dimensionalities. They report a correlation with the UMNSRS tagged for similarity of 0.46 with CBOW and 0.52 with skip-gram; and a correlation with the UMNSRS tagged for relatedness of 0.41 with CBOW and 0.45 with skip-gram.

Chiu et al. [10] evaluated both the skip-gram and CBOW word2vec models over the PMC corpus and PubMed. They evaluated a lot of hyper-parameters including vector dimensionality, windowing, learning rate, and minimum count. They used a subset of the UMNSRS dataset that ignores word pairs that do not exist in the corpus. They found that using PubMed obtained a higher correlation than PMC or a combination of PMC and PubMed. They also found that although the hyper-parameter settings can improve the performance of the model, the effects vary. They find a dimensionality of 400 and 500 for UMNSRS Sim. and Rel. are the best respectively, however their extrinsic evaluations show a dimensionality of 200 is the best. Upon closer examination, the dimensionality of 400 and 500 are almost certainly not statistically significantly different from their reported results with a vector dimensionality of 200 (n is not reported, therefore statistical significance cannot be exactly computed). They find the skip-gram model outperforms CBOW on UMNSRS.

Pakhomov et al. [11] trained CBOW word2vec over three different types of corpora: clinical (clinical notes from the Fairview Health System), biomedical (PMC corpus), and general English (Wikipedia). They evaluated the method using a subset of the UMNSRS restricting to single word term pairs. Using a window size of 8, and a dimensionality of 200, they create CBOW vectors with each corpus, and evaluate on UMNSRS. They find that the model trained on the PMC corpus obtained the highest correlation for both similarity and relatedness, with a correlation of 0.62, and 0.58 respectively. They explored varying the corpus size for the clinical data, and found that the results increased as they systematically increased the corpus size from 1 million to over 4 billion tokens. They also evaluate on several extrinsic tasks. Although techniques and parameters differ between other authors and us, our results with multi-word terms achieve comparable results to previous work evaluated on single word terms.

7. Conclusions

In this paper, we explored methods to create distributional context vectors of multi-word terms for the task of semantic relatedness. We evaluated our results on four standard evaluation datasets, MiniMayoSRS Physicians, MiniMayoSRS Coders, UMNSRS tagged for relatedness, and UMNSRS tagged for similarity, and compared against explicit co-occurrence vectors as a baseline. Dimensionality reduction was performed using singular value decomposition (SVD), word embeddings with skip-gram (SG), and word embeddings with CBOW (CBOW), and dimensionalities were varied to 100, 300, 500, 1000, and 1500 for each dimensionality reduction techniques. Additionally dimensionalities of 2000, 2500, and 3000 we evaluated for SVD. Favoring lower dimensional vectors, we found that vector dimensionality of 200 is best for SG and CBOW, and a dimensionality of 1000 is best for SVD. We found that SG and CBOW created better vector representations than explicit and SVD, but their is no significant increase in correlation using SG versus CBOW. We also compared the performance of multi-word term aggregation methods of summation of component word vectors, averaging of component word vectors, creating multi-word term vectors using the compoundify tool, and creating concept vectors using the MetaMap tool. Using concept vectors achieved the highest sum of correlations across all datasets, but only marginally. We found no statistical significance between any multi-word term aggregation method across all dimensionality reduction techniques, and dimensions tested. This suggests that multi-word term aggregation methods of sum or mean are sufficient to quantify the relatedness between multi-word terms without preprocessing (MetaMap or compoundify).

Conflict of interest statement

We have no conflicts of interest.

References

[1] R. Rada, H. Mili, E. Bicknell, M. Blettner, Development and application of a metric on semantic nets, IEEE Trans. Syst., Man, Cybernet. 19 (1) (1989) 17–30.
[2] Y. Lin, W. Li, K. Chen, Y. Liu, A document clustering and ranking system for exploring MEDLINE citations, J. Am. Med. Inform. Assoc. 14 (5) (2007) 651–661.
[3] O. Bodenreider, A. Burgun, Aligning knowledge sources in the UMLS: methods, quantitative results, and applications, in: Proceedings of the 11th World Congress on Medical Informatics (MEDINFO), San Fransico, CA, 2004, pp. 327–331.
[4] T. Pedersen, S. Pakhomov, S. Patwardhan, C. Chute, Measures of semantic similarity and relatedness in the biomedical domain, J. Biomed. Inform. 40 (3) (2007) 288–299.
[5] J. Weeds, D. Weir, D. McCarthy, Characterising measures of lexical distributional similarity, Proceedings of the 20th international conference on Computational Linguistics, Association for Computational Linguistics,, 2004, p. 1015.
[6] W.-T. Yih, V. Qazvinian, Measuring word relatedness using heterogeneous vector space models, Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, 2012, pp. 616–620.
[7] J. Reisinger, R.J. Mooney, Multi-prototype vector-space models of word meaning, Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2010, pp. 109–117.
[8] K. Radinsky, E. Agichtein, E. Gabrilovich, S. Markovitch, A word at a time: computing word relatedness using temporal semantic analysis, Proceedings of the 20th International Conference on World Wide Web, ACM, 2011, pp. 337–346.
[9] T. Muneeb, S.K. Sahu, A. Anand, Evaluating distributed word representations for capturing semantics of biomedical concepts, Proc. ACL-IJCNLP (2015) 158.
[10] B. Chiu, G. Crichton, A. Korhonen, S. Pyysalo, How to train good word embeddings for biomedical NLP, in: Proceedings of the 15th Workshop on Biomedical Natural Language Processing, 2016, pp. 166–174.
[11] S. Pakhomov, G. Finley, R. McEwan, Y. Wang, G. Melton, Corpus domain effects on distributional semantic modeling of medical terms, Bioinformatics 32 (2016) 3635–3644.
[12] S. Patwardhan, T. Pedersen, Using WordNet-based context vectors to estimate the semantic relatedness of concepts, in: Proceedings of the EACL 2006 Workshop Making Sense of Sense – Bringing Computational Linguistics and Psycholinguistics Together, Trento, Italy, 2006, pp. 1–8.
[13] H. Schütze, Dimensions of meaning, in: Proceedings of the ACM/IEEE Conference on Supercomputing, Minneapolis, MN, 1992, pp. 787–796.
[14] Y. Liu, B. McInnes, T. Pedersen, G. Melton-Meaux, S. Pakhomov, Semantic relatedness study using second order co-occurrence vectors computed from biomedical corpora, UMLS and WordNet, Proceedings of the 2nd ACM SIGHIT Symposium on International Health Informatics, ACM, 2012, pp. 363–372.
[15] A. Sajadi, E. Milios, V. Kešelj, J. Janssen, Domain-specific semantic relatedness from wikipedia structure: a case study in biomedical text, Computational Linguistics and Intelligent Text Processing, vol. 9041, Springer International Publishing, 2015, pp. 347–360.
[16] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, R. Harshman, Indexing by latent semantic analysis, J. Am. Soc. Informat. Sci. 41 (6) (1990) 391.
[17] T. Pedersen, Unsupervised corpus-based methods for WSD, Word sense disambiguation: algorithms and applications, 2006, pp. 33–166.
[18] A. Sabbir, A. Yepes, R. Kavuluru, Knowledge-based biomedical word sense disambiguation with neural concept embeddings and distant supervision, Available from: < 1610.08557 > .
[19] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in Neural Information Processing Systems, 2013, pp. 3111–3119.
[20] A.R. Aronson, F.-M. Lang, An overview of metamap: historical perspective and recent advances, J. Am. Med. Inform. Assoc. 17 (3) (2010) 229–236.
[21] T. Pedersen, S. Banerjee, B. McInnes, S. Kohli, M. Joshi, Y. Liu, The Ngram statistics package (Text::NSP): A flexible tool for identifying ngrams, collocations, and word associations, Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World, Association for Computational Linguistics, 2011, pp. 131–133.
[22] S. Pakhomov, T. Pedersen, B. McInnes, G. Melton, A. Ruggieri, C. Chute, Towards a framework for developing semantic relatedness reference standards, J. Biomed. Inform. 44 (2) (2011) 251–265.
[23] S. Pakhomov, B. McInnes, T. Adam, Y. Liu, T. Pedersen, G. Melton, Semantic similarity and relatedness between clinical terms: An experimental study, in: Proceedings of the American Medical Informatics Association (AMIA) Symposium, Washington, DC, 2010, pp. 572–576.
[24] R.A. Fisher, Frequency distribution of the values of the correlation coefficient in samples from an indefinitely large population, Biometrika (1915) 507–521.