DRCoVe: An Augmented Word Representation Approach using Distributional and Relational Context

Md. Aslam Parwez
Dept. of Computer Sc.
Jamia Millia Islamia
New Delhi, India
aslamparwez.jmi@gmail.com

Muhammad Abulaish
Dept. of Computer Sc.
South Asian University
New Delhi, India
abulaish@sau.ac.in

Mohd. Fazil
Dept. of Computer Sc.
South Asian University
New Delhi, India
mohdfazil.jmi@gmail.com

Abstract

Word representation using the distributional information of words from a sizeable corpus is considered efficacious in many natural language processing and text mining applications. However, distributional representation of a word is unable to capture distant relational knowledge, representing the relational semantics. In this paper, we propose a novel word representation approach using distributional and relational contexts, DRCoVe, which augments the distributional representation of a word using the relational semantics extracted as syntactic and semantic association among entities from the underlying corpus. Unlike existing approaches that use external knowledge bases representing the relational semantics for enhanced word representation, DRCoVe uses typed dependencies (aka syntactic dependencies) to extract relational knowledge from the underlying corpus. The proposed approach is applied over a biomedical text corpus to learn word representation and compared with GloVe, which is one of the most popular word embedding approaches. The evaluation results on various benchmark datasets for *word similarity* and *word categorization* tasks demonstrate the effectiveness of DRCoVe over the GloVe.

1 Introduction

Understanding contextual semantics of words is crucial in many natural language processing (NLP) applications. Recent trends in text mining and NLP suggest immense interest towards learning word embedding or word representation in a vector space from a large corpus, which could be useful for a variety of applications like text classification (Lai et al., 2015), clustering (Wang et al., 2015), and sentiment analysis (Tang et al., 2014). In addition, researchers are devising methods to learn phrase-, sentence-, or document-level embeddings for various NLP applications. Word embeddings capture implicit semantics and hence attracted many researchers to explore and exploit a tremendous amount of available unstructured corpora for efficient word representation by employing mainly unsupervised learning approaches. Further, the growth and availability of domain-specific massive text corpora can be exploited to learn domain-specific word representation.

Although different approaches for learning word embeddings have been proposed in the prior works, they are mostly based on distributional representation of words, considering the neighbors of a word within a fixed context window. These algorithms map sparse representation of words to a lower dimensional vector space where words with similar context appear nearby each other. However, distributional representation of words learned by these algorithms suffer from two important limitations – (i) unable to capture the relational semantics of rare co-occurring words within the corpus, and (ii) unable to capture the relational semantics of words that are outside the purview of the context window. The first limitation is that a large corpus, though represents different contextual information, may have rare co-occurrence of two words because it might not be large enough to possess sufficient count of the co-occurrence of semantically similar word pairs. To overcome this limitation, researchers have incorporated knowledge into these distributional word representations from external knowledge bases (KBs). In this direction, semantically related words in terms of relations like *synonymy*, *hypernymy*, and *meronymy* from KBs like WordNet (Miller, 1995), Freebase (Bollacker et al., 2008) have been used to learn better representation of words (Alsuhaibani et al.,

D M Sharma, P Bhattacharyya and R Sangal. Proc. of the 16th Intl. Conference on Natural Language Processing, pages 220–229
Hyderabad, India, December 2019.

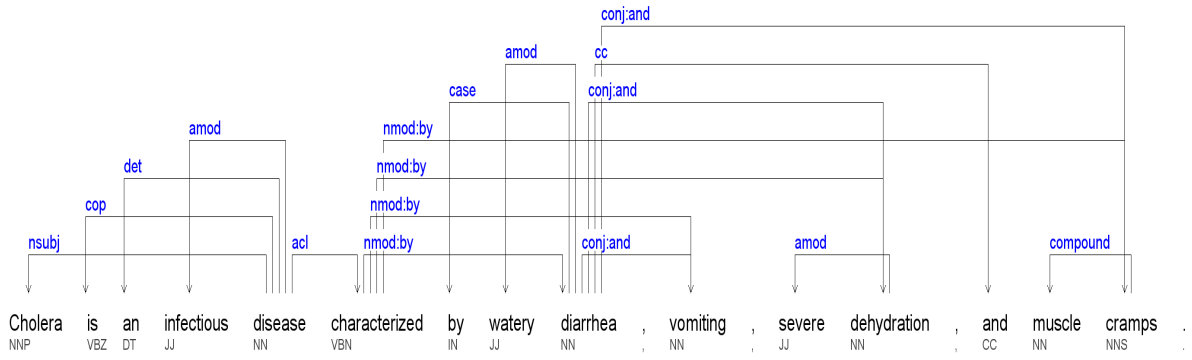

Figure 1: An exemplar dependency parse tree generated by the Stanford parser using DependenSee 3.7.0

2018; Celikyilmaz et al., 2015). This makes these approaches dependent upon the external KBs to enhance the efficacy of word representation. Although KBs provide significant information about word relations, they are scanty with limited entries for each word and does not represent any contextual information. In addition, since KBs are manually curated and maintained, they are not comprehensive.

The second limitation is that the distributional word representations are unable to capture the relational semantics of words due to their dependence on the fixed context window, and hence ignore the semantic associations between words that are outside the purview of the context window. For example, in the sentence, *"cholera is an infectious disease characterized by watery diarrhea, vomiting, severe dehydration, and muscle cramps"*, the word pairs (*cholera*, *dehydration*) and (*cholera*, *cramps*) have long-range dependency. However, both *dehydration* and *cramps* are semantically associated with *cholera* as they are its symptoms. In case of fixed context window size, e.g. 5, such long range dependency relationships will not be captured. Further, if we increase the size of the context window, it will adversely impact the embedding representation due to the inclusion of irrelevant and weak contextual words. Additionally, in case of domain-specific corpus for learning word embedding, the semantic relation between *cholera* and *dehydration*, or *cholera* and *cramps* would be very vital because *dehydration* and *muscle cramps* are the symptoms of *cholera*. These relational semantics can be captured by dependency grammar that shows syntactic and semantic relationships between words of a sentence. To this end, Levy and Goldberg (2014a) presented a dependency-based word representation learning approach to incorporate the *syntactic contexts* instead of *linear contexts*. However, existing literatures have no approach that learn word representation using syntactic contexts extracted from inter-relationships of words based on the dependency tuples generated by the language-parser. For example, in figure 1, the syntactic contexts using only the head and modifier words of the dependency tuples generated by the parser shows direct dependency relation between *cholera* and *disease* through *nsubj* dependency relation; but, it doesn't not show any relational semantics between *cholera* and *watery*, *diarrhea*, *vomiting*, *dehydration*, and *cramps* as they are not directly linked to *cholera* by any dependency relations. Therefore, extraction of such relations to augment word representations would be very helpful for various domain-specific NLP tasks such as classification of disease-related documents or texts. To the best of our knowledge, in the existing literatures, no such approach exists that utilizes the relational semantics extracted from a large corpus to enhance the distributional representation of words.

In this paper, we present an augmented approach, DRCoVe, to use both text corpus and an extracted repository of semantically related triplets from the corpus to learn efficient word representation. The proposed approach first initializes the word representation to low-dimensional real-valued vectors generated from the singular value decomposition (SVD) of positive pointwise mutual information (PPMI) matrix of the underlying corpus and the relational semantic repository. The initial word vectors from the corpus are augmented using vectors from the relational semantic repository, provided the words from the corpus occur in the vocabulary of the relational

semantic repository. In the proposed approach, we implement a modified GloVe (Pennington et al., 2014) objective function for cost optimization to incorporate vector representations from the relational knowledge repository with the initial vectors from the corpus. In brief, the main contributions of this paper can be summarized as follows.

- We propose DRCoVe, a novel approach of learning and augmentation of word representation from a corpus that can handle both long- and short-range dependencies among words.
- The model combines the benefits of point-wise mutual information, singular value decomposition, and neural network-based updation.
- Compared to existing approaches, the proposed model performs considerably better on different benchmark datasets.

Rest of the paper is organized as follows. Section 2 presents a brief review of the existing works on learning word representations. Section 3 presents background details of the concepts used in this paper. Section 4 presents the detailed description of the proposed model. Section 5 presents the experimental details and evaluation results. Finally, section 6 concludes the paper and provides future directions of research.

2 Related Works

Recently, a number of different learning algorithms have been proposed to learn the low-dimensional dense representation of words generally called word embedding used in different NLP tasks such as named entity recognition (Collobert et al., 2011), sentiment analysis (Tang et al., 2014). In this regard, two popular word representation models: continuous bag of words (CBOW) and skip gram (SG) (Mikolov et al., 2013a) models based on neural networks have gained momentum in learning distributed word representation by exploiting the local context of words co-occurring within a given context window. The CBOW predicts the target word given the surrounding context words while SG predicts the surrounding context words given the current word. Similarly, GloVe (Pennington et al., 2014) is another popular method of learning word representation based on global co-occurrence matrix that predicts global co-occurrence between target and context words by employing randomly initialized vectors of desired dimensions. These models learn embeddings only from the corpus without incorporation of any external knowledge. However, in this direction, numerous studies (Yu and Dredze, 2014; Xu et al., 2014; Alsuhaibani et al., 2018) have attempted to incorporate the relational information from KBs for word representation. In Yu and Dredze (2014), the authors proposed an approach to jointly learn embeddings from a corpus and a similarity lexicon (synonymy) by assigning high probabilities to words that appear in the similarity lexicon using joint objective functions of relation constraint models (RCM) and CBOW. Similarly, Xu et al. (2014) used the relational and categorical information as regularization parameters to the SG training objective function to improve the word representation. The CBOW based models normalize target word probabilities for the whole vocabulary, hence, computationally very expensive for large corpora.

In Ghosh et al. (2016), the authors proposed vocabulary driven skip-gram with negative sampling (SGNS) to learn disease-specific word vectors from health-related news corpus by incorporating disease-related vocabulary. Most of the proposed word representation approaches are based on either of the two models (CBOW or SG), or their variants (SGNS, SGHS) of Word2Vec algorithm either by linearly combining additional objective functions or adding as regularizers. Alsuhaibani et al. (2018) used WordNet to extract eight different types of relations such as *synonymy*, *antonymy*, *hypernymy*, *meronymy*, and so on to learn joint embeddings. They used a linear combination of GloVe and KB-based objective functions. All the discussed and existing approaches ignore the relational semantics between the words, which are outside the purview of context-window.

3 Background and Problem Definition

This section presents the notations and the background details of the important concepts used in the proposed approach.

Notations: Suppose a corpus $\mathcal{C}$ has n number of documents $d_1, d_2, \ldots, d_n$, and D represents the collection of target and context words pairs (w, c) obtained from $\mathcal{C}$ for a given context

window size l, where context words of a target word w_i are the surrounding words $w_{i-l}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+l}$. In addition, assume that V_w and V_c represent word and context vocabularies respectively for corpus D. We assume that $n_{(w,c)}$ represents the total count of (w, c) pair in D such that the target word w and context word c appear together within the context window l, n_w and n_c denote the occurrence of w and c respectively in D such that $n_w = \sum_{\hat{c} \in V_c} n_{(w,\hat{c})}$ and $n_c = \sum_{\hat{w} \in V_w} n_{(\hat{w},c)}$. The association between every pair of target and context words of V_w and V_c is presented in a matrix M such that each row of the matrix represents the vector of a target word $w \in V_w$ and each column represents vector of a context word $c \in V_c$ and every element $M_{i,j}$ represents the association between the i^{th} target word w_i and j^{th} context word c_j. Further, assume a relational semantic repository R_l consisting of all the relational semantic triplets extracted from the corpus $\mathcal{C}$. In addition, assume $\mathcal{V}$ represents the vocabulary of R_l. In the paper, alphabets w and c in bold typeface represent vectors.

GloVe: It is a neural network-based machine learning algorithm to learn an efficient lower dimensional dense representation of words in an embedding space. It uses global co-occurrence matrix to learn distributed representation of words from a text corpus. Initially, it creates co-occurrence matrix M with rows representing target words for which we want to learn word representation and the columns represent the context words co-occurring with the target words in the corpus within a given context window. In M, each entry, say, $M_{i,j}$ represents the sum of the reciprocal of the distance of co-occurring target and context words. `GloVe` implements weighted least square regression objective function to minimize the loss J_g as given in equation 1, where $f(M_{w,c})$ is the weight function to find weight between a target word w and context word c as given in equation 2, and b_w and b_c are the bias terms for the underlying target and context words respectively. In the equation 2, $\alpha = 0.75$ is a hyper-parameter and $x_{max} = 100$. The objective of `GloVe` is to minimize the squared difference between the inner product of word and context vectors **w** and **c**, and the logarithm of their co-occurrence count in D.

$$J_g = \frac{1}{2} \sum_{w \in V_w} \sum_{c \in V_c} f(M_{w,c})(\mathbf{w}^T \cdot \mathbf{c} + b_w + b_c - log(M_{i,j}))^2 \tag{1}$$

$$f(M_{w,c}) = min\left\{(M_{w,c}/x_{max})^{\alpha}, 1\right\} \tag{2}$$

In `GloVe`, learning process starts by assigning random vectors of desired dimensions to the target and context words and then updating them during the learning process with an objective to reduce the weighted least square loss as given in equation 1.

Pointwise Mutual Information: In the existing literature, researchers have used different metrics such as co-occurrence count in `GloVe` to represent the association between a word and context pair (w, c). However, simple frequency count is not the best measure of association as it does not incorporate any contextual information. The pointwise mutual information (PMI) is another measure of association and better as compared to co-occurrence count. It measures how often two events co-occur compared to what we would expect if they were independent as defined in equation 3 (Jurafsky and Martin, 2018). There can be target and context word pairs ($w \in V_w$ and $c \in V_c$) which do not appear together within the given context window l in the corpus and for such pairs $n_{(w,c)} = 0$, and therefore $PMI(w, c) = log(0) = -\infty$. To avoid this situation, positive pointwise mutual information (PPMI) has been used in which negative `PMI` values are mapped to zero as given in equation 4. In addition, Bullinaria and Levy (2007) showed that `PPMI` performs better than `PMI` in finding semantic similarity. `PPMI` measures are widely used to find semantic similarity, however, these matrices are highly sparse and need huge computational resources. One measure is to convert such sparse vectors into low dimensional dense vectors to improve computational efficiency and generalization. In this regard, dimensionality reduction is a way to find low dimensional dense vectors using matrix factorization techniques such as `SVD`.

$$PMI(w, c) = log\left(\frac{P(w, c)}{P(w) * P(c)}\right) = log\left(\frac{n_{(w,c)} * |D|}{n_w * n_c}\right) \tag{3}$$

$$PPMI(w, c) = max\left\{PMI(w, c), 0\right\} \tag{4}$$

Singular Value Decomposition: It is a dimensionality reduction method which decomposes a symmetric matrix $M_{m\times n}$ into three matrices U, Σ, and V such that $M = U \cdot \Sigma \cdot V$. The matrices U and V are orthogonal matrices while Σ is a diagonal matrix of singular values. To obtain d dimensional vectors, the matrix M is decomposed to $U_{m\times d}$, $\Sigma_{d\times d}$, and $V_{d\times n}$ corresponding to top d singular values. The d-dimensional rows of matrix $W = U \cdot \sqrt{\Sigma}$ are dense vectors which are the approximate representative of high dimensional rows of M. The matrix W is considered as a dense vector representation of words, while the matrix $C = V^T \cdot \sqrt{\Sigma}$ can be considered as context representation. The matrices W and C thus obtained are used as initial word and context representations respectively. These resulting representations need to fulfill minimization of error in matrix decomposition.

4 Proposed Approach

This section presents the detailed description of the proposed approach, starting from the mechanism to generate initial word representation from the corpus, their augmentation through relational semantics, and finally, adaptive updation of word vectors. A detailed description of each step of the proposed approach is presented in the following subsections.

4.1 Initial Vector Representation

To learn word representation of desired dimension, we first need to initialize the vectors for each target and context words pair of the corpus that can be further augmented using their relational semantics and updated based on the weighted least square loss minimization process. Before neural-based approaches, distributed word representations were based on count-based vectors such as *tf-idf* and *SVD-based* vectors. Recent advancements in neural network-based word representation have shown significant improvement in its performance in various NLP tasks. The neural network-based word representations are based on prediction (Mikolov et al., 2013b,a) of either the target word given the context within the specified context window or vice versa. However, recent studies (Levy and Goldberg, 2014b; Levy et al., 2015) have shown that the neural network-based embedding learned using Word2Vec or GloVe models are comparable in performance with the traditional representation of vectors obtained through the decomposition of PPMI matrix. Therefore, to incorporate the benefits of traditional decomposition-based vectors, the proposed approach generates initial word representation using vectors obtained from SVD-based factorization of PPMI matrix. To this end, we first create a co-occurrence matrix M considering the co-occurrence count of every (w, c) pair of target and context words from V_w and V_c respectively that is further mapped to a PPMI matrix M_p. Thereafter, the M_p is factorized using SVD to generate initial low dimensional dense vector representations of target and context words as $W = U \cdot \sqrt{\Sigma}$ and $C = V^T \cdot \sqrt{\Sigma}$, respectively from the corpus that incorporate the distributional semantics. Similarly, the same process is repeated for relational semantic repository R_l to generate the initial vector representation of target and context words as $\hat{W} = U \cdot \sqrt{\Sigma}$ and $\hat{C} = V^T \cdot \sqrt{\Sigma}$, respectively from R_l. The initial vectors of target and context words from the corpus are further augmented using the vectors generated from the relational semantic repository. A detailed description of the augmentation process is described in the following section.

4.2 Objective Function Augmentation

To minimize the decomposition error we followed the GloVe approach of optimization of the initial representation of vectors. GloVe method learns continuous word representation from a corpus using the global co-occurrence matrix. However, Glove does not incorporate any additional or domain-specific knowledge and suffers from two important limitations as discussed in section 1. Therefore, during optimization we performed the augmentation of initial word representation from the corpus by merging with the initial word representation from the relational semantic repository. To augment the additional information during learning, we define an augmented objective function J_a similar to GloVe as given in equation 5, where $f(p_{w,c})$ is the weight function to assign weight between every pair (w, c) of target and context words as given in equation 6, and b_w and b_c are the bias values for w and c respectively, and $p_{w,c}$ is the PPMI value between w and c. In equation 6, α is a hyper parameter and we used

0.75 as its value as used in GloVe.

$$J_a = \frac{1}{2} \sum_{w \in V_w} \sum_{c \in V_c} f(p_{w,c})(\mathbf{w'}^T \cdot \mathbf{c'} + b_w + b_c - log(p_{w,c}))^2 \tag{5}$$

$$f(p_{w,c}) = min \left\{ \left(p_{w,c} / \max_{\forall w,c \in D} (p_{w,c}) \right)^{\alpha}, 1 \right\} \tag{6}$$

Thereafter, we employed the relational semantics from the extracted relational semantic repository R_l consisting of vocabulary $\mathcal{V}$ to augment the learning process. The input corpus $\mathcal{C}$ consist of target and context words pairs $(w, c) \in D$. Based on $\mathcal{V}$, we grouped the (w, c) pairs of D into three categories – $D_\wedge$, $D_\sim$, and $D_\oplus$ such that

- $D_\wedge = \{(w, c) : w \in \mathcal{V} \wedge c \in \mathcal{V}\}$, i.e. both the target and context words belongs to $\mathcal{V}$
- $D_\sim = \{(w, c) :\sim (w \in \mathcal{V} \wedge c \in \mathcal{V})\}$, i.e. neither target nor the context word belongs to $\mathcal{V}$
- $D_\oplus = \{(w, c) : w \in \mathcal{V} \oplus c \in \mathcal{V}\}$, i.e either the target or the context word belongs to $\mathcal{V}$

We need to consider each of these (w, c) pair categories especially while merging to generate augmented word representation.

In case of $D_\wedge$, as both the target and context words belong to $\mathcal{V}$, we considered the merged vectors from the corpus and the relational semantic repository corresponding to target and context words such that $\mathbf{w'} = 0.5 * (\mathbf{w} + \hat{\mathbf{w}})$ and $\mathbf{c'} = 0.5 * (\mathbf{c} + \hat{\mathbf{c}})$, where $\mathbf{w}$ and $\mathbf{c}$ are the initial vectors from corpus and $\hat{\mathbf{w}}$ and $\hat{\mathbf{c}}$ are the initial vectors from relational semantic repository. For category $D_\sim$, we considered the initial vectors from the corpus only as neither of the two words belongs to $\mathcal{V}$, hence, we have $\mathbf{w'} = \mathbf{w}$ and $\mathbf{c'} = \mathbf{c}$. Similarly, in case of $D_\oplus$, as either of the two words belongs to $\mathcal{V}$ but not both, we took the merged vector for target or context word depending upon which word belongs to $\mathcal{V}$. In this case, if target word belongs to $\mathcal{V}$, we take $\mathbf{w'} = 0.5 * (\mathbf{w} + \hat{\mathbf{w}})$ and if context word belongs to $\mathcal{V}$, we consider $\mathbf{c'} = 0.5 * (\mathbf{c} + \hat{\mathbf{c}})$.

4.3 Adaptive Updation of Parameters

We performed the parameter updation during learning process based on a well-known gradient descent technique called AdaGrad (Duchi et al., 2011), which is an adaptive gradient update algorithm to perform gradient-based learning. The gradients are computed as follows:

Table 1: Concept categorization performance with $l = 5$, and $d = 100$

Word Embeddings	AP	BLESS	Battig	ESSLI_1a	ESSLI_2b	ESSLI_2c
GloVe_W	0.1940	0.21	0.0999	0.4090	**0.575**	0.3333
GloVe_Merged	**0.2213**	0.21	0.1062	0.4318	0.55	0.3555
DRCoVe_W	0.1890	0.235	0.0995	0.4318	0.45	0.377
DRCoVe_C	0.1990	**0.26**	0.1062	**0.4772**	0.475	**0.4222**
DRCoVe_Merged	0.1965	0.245	**0.1081**	0.4545	0.5	0.4

Table 2: Concept categorization performance with $l = 5$, and $d = 200$

Word Embeddings	AP	BLESS	Battig	ESSLI_1a	ESSLI_2b	ESSLI_2c
GloVe_W	0.1815	0.205	0.0982	0.4318	**0.55**	0.3777
GloVe_Merged	0.2039	0.225	0.1049	**0.4545**	0.525	0.3777
DRCoVe_W	0.1940	**0.23**	0.1013	0.4090	0.475	0.3777
DRCoVe_C	0.2064	0.215	**0.1060**	0.4090	0.475	0.3777
DRCoVe_Merged	**0.2068**	0.225	0.1009	0.4318	0.45	**0.4**

$$\frac{\delta J}{\delta \mathbf{w'}} = g_{t,\mathbf{w'}} = \sum_{c \in V_c} f(p_{w,c})(\mathbf{w'}^T \cdot \mathbf{c'} + b_w + b_c - log(p_{w,c})) \cdot \mathbf{c'} \tag{7}$$

$$\frac{\delta J}{\delta \mathbf{c'}} = g_{t,\mathbf{c'}} = \sum_{w \in V_w} f(p_{w,c})(\mathbf{w'}^T \cdot \mathbf{c'} + b_w + b_c - log(p_{w,c})) \cdot \mathbf{w'} \tag{8}$$

$$\frac{\delta J}{\delta b_w} = g_{t,b_w} = \sum_{c \in V_c} f(p_{w,c})(\mathbf{w'}^T \cdot \mathbf{c'} + b_w + b_c - log(p_{w,c})) \tag{9}$$

$$\frac{\delta J}{\delta b_c} = g_{t,b_c} = \sum_{w \in V_w} f(p_{w,c})(\mathbf{w'}^T \cdot \mathbf{c'} + b_w + b_c - log(p_{w,c})) \tag{10}$$

AdaGrad algorithm is suitable for dealing with sparse data as it performs larger updates for infrequent words, and smaller updates for frequent words. The update equation is shown as follows:

$$\mathbf{w'}^{t+1} = \mathbf{w'}^t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,\mathbf{w'}}^2}} * (g_{t,\mathbf{w'}}) \tag{11}$$

where, $\mathbf{w'}$ is the a merged target word vector, $g_{t,w}$ is the gradient at time t, and $g_{\tau,w}^2$ is the squared gradient at time τ for the target word vector $\mathbf{w'}$. Similarly, updates for context word and biases are performed according to the following equations.

$$\mathbf{c'}^{t+1} = \mathbf{c'}^t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,\mathbf{c'}}^2}} * (g_{t,\mathbf{c'}}) \tag{12}$$

$$b_w^{t+1} = b_w^t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,b_w}^2}} * (g_{t,b_w}) \tag{13}$$

$$b_c^{t+1} = b_c^t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,b_c}^2}} * (g_{t,b_c}) \tag{14}$$

Table 3: Concept categorization performance with $l = 10$, and $d = 100$

Word Embeddings	AP	BLESS	Battig	ESSLI_1a	ESSLI_2b	ESSLI_2c
GloVe_W	0.204	0.215	0.1032	0.4091	0.525	0.3778
GloVe_Merged	0.2113	0.22	0.1095	0.4308	**0.525**	0.3778
DRCoVe_W	0.2015	**0.25**	0.0996	0.4091	0.475	0.3778
DRCoVe_C	**0.2139**	0.22	0.1017	**0.4318**	0.475	**0.4222**
DRCoVe_Merged	0.199	0.225	**0.1047**	0.4091	0.45	0.3556

Table 4: Concept categorization performance with $l = 10$, and $d = 200$

Word Embedding	AP	BLESS	Battig	ESSLI_1a	ESSLI_2b	ESSLI_2c
GloVe_W	0.1965	0.2150	0.1076	0.4090	0.55	0.4222
GloVe_Merged	**0.2313**	0.22	0.1106	0.4140	**0.625**	0.4
DRCoVe_W	0.2064	0.225	0.1026	**0.4545**	0.475	0.355
DRCoVe_C	0.1965	**0.23**	0.1085	0.4014	0.525	0.4
DRCoVe_Merged	0.2114	0.225	**0.1122**	0.4245	0.45	**0.432**

5 Experimental Setup and Results

The DRCoVe is evaluated on different benchmark datasets using two evaluation tasks – *word similarity* and *concept categorization*. This section presents a brief description of corpus and relational semantic repository used in the evaluation process, experimental setup, and finally presents the evaluation results.

5.1 Corpus and Relational Semantic Repository

The DRCoVe is evaluated on a biomedical text corpus crawled from PubMed[1], an online repository of millions of citations and abstracts related to *biomedicine*, *health*, *life and behavioral sciences*, and *bioengineering*. The abstracts are the source of rich information related to *diseases*, *symptoms*, *pathogens*, *vectors*, and their *transmission* and *etiologies*. PubMed provides access to the abstracts of documents through *axis 2.1.6.2 API*[2]. The crawled corpus $\mathcal{C}$ consist of 16,337 PubMed documents related to four diseases – *cholera*, *dengue*, *influenza*, and *malaria*. In addition, a relational semantic repository R_l is created by extracting relational triplets $< arg_1,\ relation,\ arg_2 >$ based on typed dependencies generated by Stanford parser[3] that are filtered using MetaMap[4] to identify meaningful disease-symptom triplets. The repository R_l is used to augment the learning process of word representation. We have extracted the association between the diseases and symptoms using the approach defined in (Parwez et al., 2018; Abulaish et al., 2019).

5.2 Experimental Setup

The documents of the corpus $\mathcal{C}$ are tokenized and processed by removing numbers, punctuations, and stop words. We experimented with the context window size l of 5 and 10 (i.e. for $l = 5$, the context words are the 5 preceding and 5 succeeding words to the target word) to extract the context words from the corpus. The co-occurrence matrix is created using the co-occurrence frequencies of the target and context words pair within the corpus. The co-occurrence matrix is further mapped into PPMI matrix, which is further factorized using SVD to get the initial word vector of desired dimension $d \in \{100, 200\}$. A similar procedure is repeated for relational semantic repository R_l and initial vectors are generated for the target and context words. Thereafter, initial word representation of corpus is augmented using the word representation of relational semantic repository which is then optimized using the objective function defined in equation 5. We used a stochastic gradient-based algorithm *AdaGrad* with the learning rate $\eta = 0.05$ for optimization. The proposed algorithm is executed for 50 iterations to converge into an optimum solution. As a result, we obtain two sets of enhanced embeddings, one for the target words of vocabulary V_w and another for the context words of vocabulary V_c. It has been shown that when the two embeddings of a word are combined by taking an average of the corresponding word vectors, the resultant embedding performs better (Pennington et al., 2014). We have presented results for both the word and context representation in addition to their merged representation.

5.3 Evaluation Results and Comparative Analysis

The quality of the learned word vectors based on *DRCoVe* is evaluated using *concept categorization* and *similarity prediction* tasks.

Concept Categorization: We evaluated the quality of learned word embedding based on *concept categorization*. It is the grouping of concepts from a given set of concepts into different categories. It evaluates the word representation by clustering the learned vectors into different groups. The performance is assessed based on the extent to which each cluster possesses concepts from a given category. The evaluation metric is

[1] https://www.ncbi.nlm.nih.gov/pubmed/
[2] http://axis.apache.org/axis2/java/core/
[3] http://nlp.stanford.edu/software/lex-parser.shtm
[4] https://metamap.nlm.nih.gov/

Table 5: Word similarity performance with $l = 5$, and $d = 100$

Word Embeddings	MTurk	RG65	RW	SCWS	SimLex999	TR9856	WS353	WS353R	WS353S
GloVe_W	0.1869	-0.0650	0.1881	0.27104	0.0407	0.1259	0.2288	0.1411	0.2404
GloVe_Merged	0.1976	-0.0675	**0.1891**	0.2844	0.0354	0.1275	0.2269	0.1447	0.2300
DRCoVe_W	**0.2327**	0.1726	0.1513	0.29	0.0737	0.1347	0.2881	0.2338	0.2702
DRCoVe_C	0.2049	0.1368	0.1555	0.2964	0.0780	**0.1454**	**0.2961**	**0.2467**	**0.2762**
DRCoVe_Merged	0.2270	**0.1839**	0.1284	**0.2982**	**0.0907**	0.1382	0.2690	0.2458	0.2324

Table 6: Word similarity performance with $l = 5$, and $d = 200$

Word Embeddings	MTurk	RG65	RW	SCWS	SimLex999	TR9856	WS353	WS353R	WS353S
GloVe_W	0.1915	-0.0430	0.1877	0.2837	0.0383	0.1263	0.2420	0.1542	0.2471
GloVe_Merged	0.2043	-0.0567	**0.1894**	0.2842	0.0316	0.1275	0.2314	0.1462	0.2374
DRCoVe_W	0.1919	0.087	0.1563	**0.3019**	0.0739	**0.1468**	0.2949	0.2260	0.2662
DRCoVe_C	**0.2152**	0.1336	0.1544	0.3007	0.0811	0.1405	**0.3120**	**0.2385**	**0.2966**
DRCoVe_Merged	0.2038	**0.1390**	0.1347	0.2977	**0.0915**	0.1412	0.2833	0.2272	0.250

called *purity* and it is 100% if the given standard category is reproduced completely. On the other hand, *purity* reaches to 0 when cluster quality worsens. The DRCoVe is evaluated based on *concept categorization* using 6 different benchmark datasets: *AP*, *BLESS*, *Battig*, *ESSLI_1a*, *ESSLI_2b*, and *ESSLI_2c*. The evaluation and comparison results on different combination of context window size and dimensionality over 6 benchmark datasets are given in tables 1, 2, 3, and 4 respectively. It can be observed from the tables that for concept categorization task, except *ESSLI_2b*, in most of the cases, DRCoVe embedding performs better than the GloVe embeddings.

Word Similarity: To evaluate learned vectors on *word similarity* task, we computed cosine similarity between learned embedding of word pairs and evaluated it based on average similarity rating assigned by human annotators to these word pairs from the benchmark datasets. The idea here is that the learned embeddings encapsulate semantics of the words if there is greater extent of correlation between the similarity score computed from the learned word vectors and the similarity score assigned by the human annotators. We calculated Spearman's rank correlation coefficient between the cosine similarity of learned embeddings and human rated similarity of word pairs. We used 9 different benchmark datasets – *MTurk*, *RG65*, *RW*, *SCWS*, *SimLex999*, *TR9856*, *WS353*, *WS353R*, and *WS353S* for evaluation. In addition, we also compared the quality of learned representation in terms of similarity task with the two variants of GloVe: GloVe_W and GloVe_Merged. The evaluation and comparison results on different combination of context window size and dimensionality on the benchmark datasets for *word similarity* are given in tables 5, 6, 7, and 8 respectively. On analysis of tables, it can be found that the *context* and *merged* vectors of DRCoVe are significantly better as compared to GloVe word vectors and merged vectors except RW, where GloVe is better.

6 Conclusion and Future Direction

Word embeddings learned from diverse sources using methods like GloVe as the distributional representation of words have been employed to resolve numerous natural language processing problems with considerable accuracy. However, these distributional representations are unable to capture the relational semantics of distant words and the words with rare co-occurrences in the corpus. In this paper, we have proposed DRCoVe, an augmentation approach of distributional word representations from a corpus with relational semantic information extracted from the corpus to learn enhanced word representation. We compared the proposed model based on semantic similarity and concept categorization tasks on different benchmark datasets and found that the word representation learned by DRCoVe shows better performance than the GloVe model in most of the datasets. The learned word representations could be useful for various NLP tasks like text classification or concept categorization. Learning word representations over much larger corpus and evaluation of their efficacy for short texts

Table 7: Word similarity performance with $l = 10$, and $d = 100$

Word Embeddings	MTurk	RG65	RW	SCWS	SimLex999	TR9856	WS353	WS353R	WS353S
GloVe_W	0.2223	-0.0625	0.1852	0.3013	0.0469	0.1341	0.2429	0.1776	0.2795
GloVe_Merged	0.2267	-0.0524	**0.1863**	0.3102	0.0411	0.1351	0.2393	0.1693	0.2774
DRCoVe_W	0.1930	0.0937	0.1637	0.2951	0.0538	**0.1396**	0.3179	0.2304	0.3250
DRCoVe_C	**0.2309**	0.1241	0.1639	0.2989	0.0441	0.1383	**0.3310**	**0.2506**	**0.3336**
DRCoVe_Merged	0.2085	**0.1404**	0.1393	**0.3140**	**0.0599**	0.1372	0.3033	0.2341	0.2760

Table 8: Word similarity performance with $l = 10$, and $d = 200$

Word Embeddings	MTurk	RG65	RW	SCWS	SimLex999	TR9856	WS353	WS353R	WS353S
GloVe_W	0.2209	-0.0759	0.1855	0.2943	0.0401	0.1321	0.2448	0.1755	0.2742
GloVe_Merged	0.2262	-0.0613	**0.1863**	0.3023	0.0351	0.1347	0.2398	0.1659	0.2779
DRCoVe_W	0.2040	0.0890	0.1734	**0.3162**	0.0775	**0.1425**	0.3029	0.2278	0.3101
DRCoVe_C	**0.2377**	**0.1539**	0.1755	0.3125	0.0793	0.1408	**0.3134**	**0.2364**	**0.3236**
DRCoVe_Merged	0.1848	0.1445	0.1244	0.3088	**0.0970**	0.1363	0.2545	0.2230	0.2219

like tweets classification seems one of the future directions of research.

References

Muhammad Abulaish, Md. Aslam Parwez, and Jahiruddin. 2019. Disease: A biomedical text analytics system for disease symptom extraction and characterization. *Journal of Biomedical Informatics*, 100(12):1–15.

Mohammed Alsuhaibani, Danushka Bollegala, Takanori Maehara, and Ken-ichi Kawarabayashi. 2018. Jointly learning word embeddings using a corpus and a knowledge base. *PloS one*, 13(3):1–26.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of International Conference on Manageent of Data*, pages 1247–1250, Vancouver, Canada. ACM.

John A Bullinaria and Joseph P Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39(3):510–526.

Asli Celikyilmaz, Dilek Hakkani-Tur, Panupong Pasupat, and Ruhi Sarikaya. 2015. Enriching word embeddings using knowledge graph for semantic tagging in conversational dialog systems. In *2015 AAAI Spring Symposium Series*, pages 39–42, California, USA. AAAI.

Ronan Collobert, Jason Weston, Leon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(1):2493–2537.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(1):2121–2159.

Saurav Ghosh, Prithwish Chakraborty, Emily Cohn, John S Brownstein, and Naren Ramakrishnan. 2016. Characterizing diseases from unstructured text: A vocabulary driven word2vec approach. In *Proceedings of International Conference on Information and Knowledge Management*, pages 1129–1138, Indianapolis, USA. ACM.

Daniel Jurafsky and James H. Martin. 2018. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, volume 3. Prentice-Hall, Inc.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Proceedings of International Conference on Artificial Intelligence*, pages 2267–2273, Texas, USA. AAAI.

Omer Levy and Yoav Goldberg. 2014a. Dependency-based word embeddings. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pages 302–308, Maryland, USA. ACL.

Omer Levy and Yoav Goldberg. 2014b. Neural word embedding as implicit matrix factorization. In *Proceedings of International Conference on Neural Information Processing Systems*, pages 2177–2185, Montreal, Canada. Curran Associates.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of International Conference on Neural Information Processing Systems*, pages 3111–3119, Nevada, USA. Curran Associates.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Md Aslam Parwez, Muhammad Abulaish, and Jahiruddin. 2018. Biomedical text analytics for characterizing climate-sensitive disease. *Procedia Computer Science*, 132:1002–1011.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of International Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543, Doha, Qatar. ACL.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pages 1555–1565, Maryland, USA. ACL.

Peng Wang, Jiaming Xu, Bo Xu, Cheng-Lin Liu, Heng Zhang, Fangyuan Wang, and Hongwei Hao. 2015. Semantic clustering and convolutional neural network for short text categorization. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pages 352–357, Beijing, China. AAAI.

Chang Xu, Yalong Bai, Jiang Bian, Bin Gao, Gang Wang, Xiaoguang Liu, and Tie-Yan Liu. 2014. Rc-net: A general framework for incorporating knowledge into word representations. In *Proceedings of International Conference on Information and Knowledge Management*, pages 1219–1228, Shanghai, China. ACM.

Mo Yu and Mark Dredze. 2014. Improving lexical embeddings with semantic knowledge. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, pages 545–550, Maryland, USA. ACL.