Integrating Extra Knowledge into Word Embedding Models for Biomedical NLP Tasks

Yuan Ling*†, Yuan An*, Mengwen Liu*, Sadid A. Hasan†, Yetian Fan‡ and Xiaohua Hu*
*College of Computing & Informatics
Drexel University, Philadelphia, PA 19104
e-mail: {yl638,ya45,ml943,xh29}@drexel.edu
†Artificial Intelligence Laboratory, Philips Research North America, Cambridge, MA
e-mail: sadid.hasan@philips.com
‡School of Mathematical Sciences, Dalian University of Technology, Dalian, China 116023

***Abstract*—Word embedding in the NLP area has attracted increasing attention in recent years. The continuous bag-of-words model (CBOW) and the continuous Skip-gram model (Skip-gram) have been developed to learn distributed representations of words from a large amount of unlabeled text data. In this paper, we explore the idea of integrating extra knowledge to the CBOW and Skip-gram models and applying the new models to biomedical NLP tasks. The main idea is to construct a weighted graph from knowledge bases (KBs) to represent structured relationships among words/concepts. In particular, we propose a GCBOW model and a GSkip-gram model respectively by integrating such a graph into the original CBOW model and Skip-gram model via graph regularization. Our experiments on four general domain standard datasets show encouraging improvements with the new models. Further evaluations on two biomedical NLP tasks (biomedical similarity/relatedness task and biomedical Information Retrieval (IR) task) show that our methods have better performance than baselines.**

I. Introduction

Distributed word representations for solving NLP problems have attracted much attention over the years [1], [2], [3], [4], [5], [6], [7]. In contrast to traditional one-hot representation, which has the limitation of representing implied semantic relations among words, distributed representation uses a dense and low dimensional vector to represent a word. In this scenario, similar words are transferred into similar vector representations by capturing semantic information among words. Mikolov et al. [8], [9] proposed two embedding methods: continuous bag-of-words model (CBOW) and continuous Skip-gram model (Skip-gram) to attract a great deal of attention among NLP researchers and practitioners [10], [11], [12].

However, embedding models have certain limitations. The unlabeled text corpus may contain noises for learning. For example, words may have incomplete and ambiguous meanings. Recently, some researchers have attempted to encode extra knowledge into word embedding models [13], [14], [15], [16], [17], [18], [19], [20]. One frequently mentioned knowledge resource for enhancing word embedding models is the structured Knowledge Base (KB). We have witnessed a quick development of KBs in past years. KBs store structured information about entity types and relation triples. A triple is represented as $\langle subject, predicate, object \rangle$. Many large-scale KBs of general or specific domains have been constructed, such as WordNet [21], Yago [22], Freebase[23], DBpedia [24], NELL [25], and UMLS [26]. KBs are useful resources and powerful tools for supporting NLP tasks such as relation extraction [27], [28] and question answering [29].

In the biomedical domain, there is a growing number of studies on applying word embedding models to biomedical NLP tasks. Tang et al. [30] studied the effect of word embedding features on biomedical named entity recognition tasks. Muneeb et al. [31] evaluated two word embedding models: word2vec and GloVe on a word similarity task. The effect of input corpus and different kinds of parameters for word embedding models are systematically evaluated on biomedical NLP tasks [32], [33], [34]. The parameters include negative sample size, learning rate, vector dimension, context window size etc. In spite of the fact that KBs play an important role for biomedical NLP tasks [35], [36], to the best of our knowledge, there is little work on integrating KBs with word embedding models for biomedical NLP tasks.

In this paper, we explore the idea of using extra knowledge from KBs to improve word embedding models for biomedical NLP tasks. We propose a Graph regularized CBOW (GCBOW) model and a Graph regularized Skip-gram (GSkip-gram) model. GCBOW and GSkip-gram models use a graph to represent knowledge from KBs and integrate the graph regularization to basic CBOW and Skip-gram models, respectively. The proposed models can be easily adopted to different types of KBs. In addition, we apply two different distance metrics for the graph regularization framework. Inspired by the contradictory results of applying word embedding to different tasks discussed in [33], we conduct experiments on both general domain tasks for intrinsic evaluation and biomedical NLP tasks for extrinsic evaluation. We evaluate our models on four general domain word similarity datasets: TOEFL synonym dataset, WordSimilarity-203, RG65, and SimLex-999. The results show that our models achieve promising improvement in precision on TOEFL synonym dataset and spearman's ρ score on other three datasets. Furthermore, we evaluate the models on two biomedical NLP tasks: biomedical concept similarity/relatedness task and biomedical Information Retrieval (IR) task. Our method achieves better scores than the baselines for both tasks.

Our major discoveries in this work are summarized below:

- Integrating extra knowledge can improve the performance of word embedding models. Our experiments on both general domain datasets and biomedical NLP tasks provide substantial evidence.
- For biomedical concepts similarity and relatedness tasks, GCBOW and GSkip-gram models achieve better results than baseline methods.
- Word embedding models improve the performance of biomedical IR applications through the concept weighting process. Leveraging extra knowledge from KBs improve the results.

The rest of the paper is organized as follows. Section II describes the traditional word embedding models. Section III presents our knowledge graph representation, proposes graph regularized CBOW model and Skip-gram model, and develops the parameter updating for new proposed models. Section IV describes our experimental results from intrinsic evaluation on standard datasets. Section V describes the experimental results from extrinsic evaluation on biomedical NLP tasks, and finally, Section VI concludes the paper.

II. Word Embedding Models

Word embedding models learn distributed representations of words from a large amount of unlabeled text data. Each word is represented as a dense and low-dimensional vector, and semantically similar words are transformed into similar vector representations. The CBOW and Skip-gram models are two word embedding models proposed by Mikolov et al. [8], [9].

Both CBOW and Skip-gram models are three-layer neural networks, containing input, projection, and output layers. The CBOW model learns word embedding by using context words to predict the center word w_t, where the context words refer to the neighboring words within a window size c near the center word in a sentence. Given a sequence of training words $w_1, w_2, \ldots, w_T$, the CBOW model has the following objective function:

$$J_1 = \max \frac{1}{T} \sum_{t=1}^{T} \log p(w_t | w_{t-c}, ..., w_{t-1}, w_{t+1}, ..., w_{t+c}) \tag{1}$$

The Skip-gram model predicts surrounding words $w_{t-c}, ..., w_{t-1}, w_{t+1}, ..., w_{t+c}$ given the current center word w_t. This model has the following objective function:

$$J_2 = \max \frac{1}{T} \sum_{t=1}^{T} \sum_{-c \le j \le c, j \ne 0} \log p(w_{t+j} | w_t) \tag{2}$$

The probability $p(w_t|w_{t+j})$ is calculated using a softmax function:

$$p(w_t | w_{t+j}) = \frac{\exp(v_t^{'T} v_{t+j})}{\sum_{n=1}^{N} \exp(v_n^{'T} v_{t+j})}, \tag{3}$$

where v_n and $v_n^{'}$ are the input and the output representation vectors of word w_n, and N is the total vocabulary size. Note that, the representation vectors v_n are between input layer and projection layer, and $v_n^{'}$ are between projection layer and output layer.

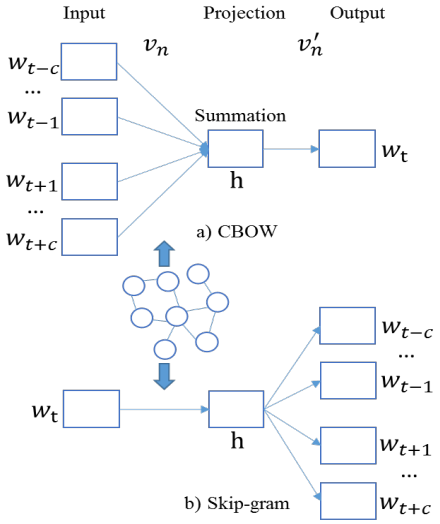

Fig. 1. Word embedding models with graph regularization.

In the CBOW model, the projection layer h is the average value of the input representations of context words.

$$h = \frac{1}{2c} \sum_{-c \le j \le c, j \ne 0} v_{t+j} \tag{4}$$

In the Skip-gram model, the projection layer h is the same as the input representation of word w_t, which is v_t.

III. Graph Regularized Embedding Models

We use an undirected graph to represent knowledge from structured KBs. Relations between words from KBs can be represented as weighted edges between word nodes in the graph. We assume embedding representations of two words should be able to represent their closeness mentioned in KBs. We keep the assumption by adding a graph regularizer to the original objective function for CBOW model and Skip-gram model. The proposed graph regularization framework can use different distance metrics between words. In this study, we explore two specific distance metrics to build the graph regularizer.

A. Knowledge Graph Representation

The undirected graph as displayed in Fig. 1 represents relationships among words from extra knowledge sources. Each word is represented as a node in the graph. An edge connects nodes n_i and n_j if there is a relation mentioned in KBs between two nodes. A weight value is set for each edge connected between nodes n_i and n_j. Different types of commonly used weighting schemas are discussed in the literature [37] [38]. We use a simple method to determine the weight value.

If two nodes n_i and n_j are connected because they are mentioned in KBs with similar meanings (e.g. synonym), we

set the weight value $\omega_{ij} = 1$; if they are connected with opposite meanings, we set $\omega_{ij} = -1$; if they are connected with weak similar meanings, we set $\omega_{ij} = 0.5$. Here we define weak similar meanings as two words are related but not exactly have similar meanings. For example, in WordNet, if two words are indicated as hypernym or hyponym, we assume they have weak similar meanings.

B. Graph Regularization Framework

The embedding representations of two words represent their semantic relationships. Structured KBs enhance the representation of semantic information with graph structures. Thus we introduce graph regularized CBOW and Skip-gram model for incorporating the extra knowledge. Suppose word w_i has relations with a set of other words $w_r, r \in \{1, \ldots, R\}$ in KBs. In our study, we use two types of distance metrics to measure the distance between two words w_i and w_j. Here, v_i and v_j are vector representations for word w_i and word w_j.

(1) Euclidean distance:

$$D_1(w_i, w_j) = ||v_i - v_j||_2 \tag{5}$$

(2) KL-Divergence:

$$\begin{aligned} D_2(w_i, w_j) &= \frac{1}{2}(D(w_i||w_j) + D(w_j||w_i)) \\ &= \frac{1}{2}(\sum_{k=1}^{K} v_{ik} \log \frac{v_{ik}}{v_{jk}} + \sum_{k=1}^{K} v_{jk} \log \frac{v_{jk}}{v_{ik}}) \end{aligned} \tag{6}$$

ω_{ij} stands for the weight value between word node w_i and w_j (discussed in Section III-A). By minimizing $\omega_{ij} D(w_i, w_j)$, we expect if two words have a close relation in KBs, their vector representations will also be close to each other. By adding this regularizer, we extend the original CBOW model and Skip-gram model to the proposed GCBOW and GSkip-gram models. The GCBOW model has the following objective function:

$$\begin{aligned} J_3 = \max \frac{1}{T} \sum_{t=1}^{T} (1 \\ - \lambda) \log p(w_t | w_{t-c}, ..., w_{t-1}, w_{t+1}, ..., w_{t+c}) \\ - \lambda \sum_{-c \le j \le c, j \neq 0} \sum_{r=1}^{R} \omega_{t+j,r} D(w_{t+j}, w_r) \end{aligned} \tag{7}$$

λ is a parameter to leverage the weights between the original objective and the newly added regularizer.

The GSkip-gram model has a similar objection function:

$$\begin{aligned} J_4 = \max \frac{1}{T} \sum_{t=1}^{T} (1 - \lambda) \sum_{-c \le j \le c, j \neq 0} \log p(w_{t+j} | w_t) \\ - \lambda \sum_{r=1}^{R} \omega_{tr} D(w_t, w_r) \end{aligned} \tag{8}$$

C. Parameters Updating

We use stochastic gradient descent (SGD) to maximize the objective function for the GCBOW model and GSkip-gram model.

For the representation from the projection layer to the output layer, hierarchical softmax is applied [8], [11]. Vocabulary is represented as a Huffmann binary tree. Each word w_t can be reached by a path from the root of the tree. Let $L(w_o)$ be the length of the path. $n(w_o, j)$ is the j-th unit on the path from root to word w_o, and each unit has an output vector $v'_{n(w_o,j)}$. $ch(n)$ is an arbitrary fixed child of n. $[\![x]\!] = 1$ if x is true, otherwise, $[\![x]\!] = -1$. In this path, each branch is treated as one binary classification. So $p(w_o|w_I)$ can be defined as follows:

$$p(w_o|w_I) = \prod_{j=1}^{L(w_o)-1} \sigma([\![n(w_o, j+1) = ch(n(w_o, j))]\!] v'^{T}_{n(w_o,j)} h) \tag{9}$$

For one training sample $\{w_i, w_o\}$, the training objective is $J = \max \log p(w_o|w_i)$.

By taking the derivative of J with regard to $v'_{n(w_o,j)}$, we obtain

$$\begin{aligned} \frac{\partial J}{\partial v'_{n(w_o,j)}} &= \frac{\partial J}{\partial (v'_{n(w_o,j)} h)} \frac{\partial (v'_{n(w_o,j)} h)}{\partial v'_{n(w_o,j)}} \\ &= (t_j - \frac{1}{1 + \exp(-v'_{n(w_o,j)} h)}) h \end{aligned} \tag{10}$$

$t_j = 1$, if $n(w_o, j+1) = ch(n(w_o, j))$, otherwise, $t_j = 0$.

The update equation for representation from the projection layer to the output layer is:

$$v'^{(new)}_{n(w_o,j)} = v'^{(old)}_{n(w_o,j)} + \alpha \frac{\partial J}{\partial v'_{n(w_o,j)}} \tag{11}$$

To learn the weights from input layer to projection layer, we take the derivative of J with regard to v_i:

$$\begin{aligned} \frac{\partial J}{\partial v_i} &= \sum_{j=1}^{L(w_o)-1} \frac{\partial J}{\partial (v'_{n(w_o,j)} h)} \frac{\partial (v'_{n(w_o,j)} h)}{\partial h} \frac{\partial h}{\partial v_i} \\ &= \sum_{j=1}^{L(w_o)-1} (t_j - \frac{1}{1 + \exp(-v'^{T}_{n(w_o,j)} h)}) v'_{n(w_o,j)} \frac{\partial h}{\partial v_i} \end{aligned} \tag{12}$$

After adding the graph regularizer, we also need to take the derivative $A = \sum_{r=1}^{R} (\omega_{ir} D(w_i, w_r))$ with regard to v_i:

$$\frac{\partial A}{\partial v_i} = \frac{\partial \sum_{r=1}^{R} \omega_{ir} D(w_i, w_r)}{\partial v_i} \tag{13}$$

When using D_1 distance,

$$\begin{aligned} \frac{\partial A_1}{\partial v_i} &= \frac{\partial (\sum_{r=1}^{R} \omega_{ir} D_1(w_i, w_j))}{\partial v_i} \\ &= \sum_{r=1}^{R} \omega_{ir} (v_i - v_r) \end{aligned} \tag{14}$$

When using D_2 distance,

$$\frac{\partial A_2}{\partial v_i} = \frac{\partial(\sum_{r=1}^{R} \omega_{ir} D_2(w_i, w_j))}{\partial v_i} = \sum_{r=1}^{R} \omega_{ir} \frac{1}{2}(\log \frac{v_{ik}}{v_{rk}} - \frac{v_{rk}}{v_i k} + 1) \quad (15)$$

The update equation for representation from the input layer to the projection layer is:

$$v_i^{(new)} = v_i^{(old)} + \alpha((1-\lambda)\frac{\partial J}{\partial v_i} - \lambda \frac{\partial A}{\partial v_i}) \quad (16)$$

IV. Intrinsic evaluation

We conduct thorough experiments on four standard datasets to examine whether adding graph regularization can improve the performance of word embedding models. In this intrinsic evaluation, we explore different parameter settings such as vector dimension size, window size for context words, λ value, and distance metrics. We also investigate a few examples to discuss how the models are improved by using extra knowledge from KBs. The goal of intrinsic evaluation is to evaluate our model on standard tasks for word embedding models, and find the best parameters for the biomedical NLP tasks.

A. Training Data

We train the word embedding models on the New York Times (NYT) corpus[1]. The dataset is pre-processed by sentence splitting, word tokenization, and stop words removal. We randomly sample 3M sentences from this corpus. The final training corpus contains 39,281,610 total words, and 268,032 unique words.

We use WordNet as the KB and select three types of word pairs: Similar, Antonym and Hypernym. There are 106,828-word pairs in total.

B. TOEFL Synonym Selection Task

TOEFL synonym selection task [39] contains 80 target words, and the objective is to select the correct synonym for each target word from 4 candidate words. We get vector representations from embedding models for both target word and candidate words, and use the cosine similarity to calculate a score for each target word and candidate word pair, the one with a highest score is chosen as the final answer. The evaluation metric on this task is precision, which is the total number of questions with the correct answer divided by the total number of questions.

First, we use divergence (D_2) to evaluate the distance between two words. We chose different d value and λ value to compare GCBOW with CBOW, and GSkip-gram with Skip-gram. d is the dimension size for word vector representation. We set the window size for context words $c = 5$.

Table I shows the results. We can see that when $\lambda = 5 \times 10^{-6}$, for $d = 50$ to $d = 300$, the GCBOW model has better performance than CBOW. The GSkip-gram also has better or equal performance than the Skip-gram model. When $\langle d = 50$, $\lambda = 1 \times 10^{-6}\rangle$ and $\langle d = 300$, $\lambda = 5 \times 10^{-5}\rangle$, GCBOW has worse performance than CBOW. When $\langle d = 100$, $\lambda = 1 \times 10^{-5}\rangle$ and $\langle d = 300$, $\lambda = 5 \times 10^{-5}$ or $\lambda = 5 \times 10^{-5}\rangle$, GSkip-gram has worse performance than Skip-gram. According to this result, we recommend to set $\lambda = 5 \times 10^{-6}$, and $d = 200$ for both GCBOW and GSkip-gram models.

By setting different window sizes of context words for models, we get comparison results as shown in Table II. In this experiment, we use D_2 distance, $\lambda = 5 \times 10^{-6}$, and $d = 200$. With varying windows size, GSkip-gram always has the best performance. With window sizes 3 and 5, GCBOW outperforms CBOW.

We conduct additional evaluation with Euclidean distance (D_1) compared to D_2. We set $\langle \lambda = 5 \times 10^{-6}$, $d = 200\rangle$, and $c = 5$ for this experiment. In Table III, the models with D_2 distance have better performance than models with D_1 distance. For GCBOW, D_2 distance outperforms D_1 distance by 5.1% while for GSkip-gram, D_2 distance outperforms D_1 distance by 0.4%.

C. WS203, RG65 and SimLex-999 Datasets

We use a second group of standard datasets: WordSimilarity-353 (WS353) [40], [41], RG65 [42], and SimLex-999 [43], which are frequently used for evaluating word representations. These datasets contain English word pairs along with human-assigned similarity judgments. The WS353 dataset is split into two subsets [41], one for evaluating similarity, and the other for evaluating relatedness. We use the similarity part for our experiments, which contains 203 pairs (WS203). SimLex-999 contains 999 concrete and abstract adjective, noun and verb pairs with rating scores. RG65 is a smaller set containing 65 pairs.

The evaluation metric on this task is to compare correlations (Spearman's ρ scores) between the similarity scores given by our models and those rated by human. Spearman's ρ score measures the strength of association between two ranked variables. As displayed in Table IV, GCBOW with distance D_1 outperforms CBOW. On the other hand, GSkip-gram with distance D_2 has generally better performance than the Skip-gram model.

D. Qualitative Analysis

We examine the results from TOEFL synonym selection task to understand how the GCBOW and GSkip-gram models improved the performance over the CBOW and Skip-gram models. We identified four pairs of question and correct answer, which were correctly identified by GCBOW or GSkip-gram model but missed by the original CBOW model or Skip-gram model. Our analysis showed that there were three possible reasons for the improvement on performance:

(1) Explicit relations mentioned in KBs for question and correct answer pair: For some question and correct answer pairs, there are direct relations mentioned in KBs between them. We assume that is the reason why our model, which integrates knowledge from KBs, can improve the performance.

[1]https://catalog.ldc.upenn.edu/LDC2008T19

TABLE I
PERFORMANCE (PRECISION, %) ON TOEFL SYNONYM DATASET WITH D_2 DISTANCE.

	CBOW	GCBOW			Skip-gram	GSkip-gram		
d/λ	0	5×10^{-6}	1×10^{-5}	5×10^{-5}	0	5×10^{-6}	1×10^{-5}	5×10^{-5}
50	51.9	**54.4**	50.6	51.9	53.8	53.8	**57.5**	53.8
100	54.4	**60.8**	59.5	57.0	63.8	**66.3**	52.5	63.8
200	58.2	**64.6**	62.0	60.8	66.3	68.8	**70.0**	63.8
300	58.8	**60.0**	**60.0**	56.3	68.8	**70.0**	55.0	53.8

TABLE II
PERFORMANCE (PRECISION, %) ON TOEFL SYNONYM DATASET WITH DIFFERENT WINDOW SIZES.

Window Size	CBOW	GCBOW	Skip-gram	GSkip-gram
3	57.50	**63.75**	73.75	**75.00**
5	58.20	**64.60**	66.30	**68.80**
7	**65.82**	62.03	65.00	**67.50**

TABLE III
PERFORMANCE (PRECISION, %) ON TOEFL SYNONYM DATASET WITH D_1 AND D_2 DISTANCE.

CBOW	GCBOW		Skip-gram	GSkip-gram	
	D_1	D_2		D_1	D_2
58.2	59.5	**64.6**	66.3	68.4	**68.8**

TABLE IV
PERFORMANCE (SPEARMAN'S ρ SCORES).

Dataset	CBOW	GCBOW		Skip-gram	GSkip-gram	
		D_1	D_2		D_1	D_2
WS203	0.751	**0.761**	0.745	0.655	**0.664**	0.659
RG65	0.460	**0.493**	0.466	0.548	0.457	**0.670**
Sim999	0.222	**0.242**	0.234	0.273	0.273	**0.274**

For example, the "$furnish/supply$" pair has a relation chain in KBs:

furnish→HYPERNYM←supply

(2) Implicit relations mentioned in KBs for question and correct answer pairs: Implicit relation means there are no direct relations mentioned in KBs for the question and correct answer pair, but there are indirect relations between them. For example, "$temperate/mild$" has the following indirect relation:

temperate→SIMILAR←moderate→SIMILAR←mild

Another example is "$root/origin$":

root→HYPERNYM←become→HYPERNYM
←changeOfstate→HYPERNYM←
beginning→HYPERNYM←origin

We believe these implicit relations in KBs have led to performance improvements of our model.

(3) No relations mentioned in KBs for question and correct answer pairs: In some cases, there exist no explicit or implicit relations in KBs among the question and answer words, but our models still work better. We speculate that there might be some inherent patterns discovered by the models yielding improved results.

V. EXTRINSIC EVALUATION

We adopt the best parameter settings from Section IV, and conduct experiments on two biomedical NLP tasks for the extrinsic evaluation. We examine the quality of our models on the biomedical concepts similarity/relatedness task and the biomedical IR task by comparing them against some baselines.

A. Training Data

We gather a biomedical corpus from two data sources: PubMed articles[2] and Clinical Medicine related Wikipedia articles[3]. The corpus contains over 5M sentences. We pre-process the dataset by conducting sentence splitting, word tokenization, and stop words removal. The total number of tokens is 93,095,323.

UMLS [26] is developed by the US National Library of Medicine and is a repository of biomedical vocabularies. We use the UMLS MRREL table as our KB. This table defines relationships between UMLS concepts. There

[2]https://www.ncbi.nlm.nih.gov/pmc/

[3]https://en.wikipedia.org/wiki/Category:Clinical_medicine

are over 1.6M word pairs selected from various relation types, such as disease-treatment, disease-prevention, disease-diagnosis, disease-finding, sign and symptom, causes etc. as shown in Table VII.

B. Biomedical Concepts Similarity and Relatedness

We apply our models to biomedical concepts similarity and relatedness tasks [44]. We use the two standard datasets: UMNSRS-Similarity, which is a set of 566 UMLS concept pairs manually rated for semantic similarity, and UMNSRS-Relatedness, which is a set of 588 UMLS concept pairs manually rated for semantic relatedness. We use Spearman's ρ as the evaluation metric for this task.

TABLE V
PERFORMANCE (SPEARMAN'S ρ SCORES) FOR BIOMEDICAL CONCEPTS DATASETS.

Model	UMNSRS-Similarity	UMNSRS-Relatedness
Baseline [32]	0.652	0.601
CBOW	0.755	0.734
GCBOW	**0.775**	**0.747**
Skip-gram	0.805	0.798
GSkip-gram	**0.817**	**0.807**

The results are displayed in Table V. For both datasets, GCBOW outperforms CBOW. Also, GSkip-gram has better performance than the Skip-gram model. Besides using CBOW and Skip-gram as intrinsic baselines, we also use the best results from [32] as an extrinsic baseline. Although we have a smaller corpus size (93,095,323 tokens) than the extrinsic baseline (2,721,808,542 tokens), our models obtain better scores for the biomedical concepts similarity and relatedness tasks.

C. Concept Weighting for Biomedical IR

We utilize word embedding models for a biomedical IR task through a concept weighting process. We conduct experiments on the TREC 2015 Clinical Decision Support (CDS) task [45]. The task dataset contains 30 topics, where each topic is a medical case narrative that describes patients medical history, signs/symptoms etc. The goal of this task is to return a ranked list of top 1,000 articles from a collection of biomedical literature that are relevant to answering three generic clinical questions about the diagnosis, test, and treatment. The collection of biomedical articles contains more than 733,000 articles from the PubMed Centeral (PMC)[4].

Word embedding models are involved with the concept weighting process as indicated in the following steps:

Step 1. Identifying concepts from narratives: We use MetaMap [46] to identify UMLS concepts in the case narratives. In order to avoid noises in this step, we also manually identify concepts as a comparison.

Step 2. Obtaining weights for each concept: For each concept, a vector representation is obtained from the embedding model. Each concept is measured using cosine similarity with all other concepts in order to obtain an average score. We use this score as the concept weight value applied to document retrieval. We assume that the more important concept will have a higher average score. A baseline is set by assigning a weight value of 1 for all concepts (designated as C-1).

Step 3. Retrieving relevant documents: We use the basic retrieval model, BM25 [47], and leverage the weighted concepts from *step 2* to boost the retrieval results.

We compare our models with the best performing system in TREC 2015 (designated as C-trec) [48]. The baselines used for comparison are C-1, and corresponding results generated from CBOW and Skip-gram. The evaluation measure is precision at top 5 retrieved documents (P@5).

TABLE VI
PERFORMANCE (P@5) FOR BIOMEDICAL IR TASK.

System	MetaMap Concepts	Manual Concepts
C-1	0.3033	0.3467
CBOW	0.3067	0.4200
GCBOW	0.3233	0.4233
Skip-gram	0.3633	0.4400
GSkip-gram	0.3733	**0.4667**
C-trec	0.4467	—

According to the results in Table VI, GCBOW has better performance than CBOW, and GSkip-gram also has better performance than Skip-gram. GSkip-gram with manual concepts achieves the best performance, which is better than C-1 and C-trec. Manual concept identification has better performance than using MetaMap. This means that by simply using MetaMap to identify concepts from narratives will introduce some noises.

VI. CONCLUSION

This paper presents two graph regularized word embedding models: GCBOW and GSkip-gram, which take extra knowledge from KBs into consideration. Experiments on standard word similarity tasks demonstrated that our models outperform the original CBOW and Skip-gram models correspondingly. We adopted the best parameter settings from the standard datasets evaluation and applied the models to two biomedical NLP tasks. Experimental results showed that integrating extra knowledge improved the performance for these two biomedical NLP tasks. Our models also demonstrated better results than baselines in these tasks.

[4]http://www.trec-cds.org/2015.html#documents

REFERENCES

[1] G. E. Hinton, J. L. McClelland, and D. E. Rumelhart, "Distributed representations," *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1*, pp. 77–109, 1986.

[2] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *journal of machine learning research*, vol. 3, no. Feb, pp. 1137–1155, 2003.

[3] A. Mnih and G. Hinton, "Three new graphical models for statistical language modelling," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 641–648.

TABLE VII
RELATION TYPES FROM UMLS MRREL.

Relation Category	Relation Type
Disease-treatment	disease_has_accepted_treatment_with_regimen
	may_be_treated_by
	may_treat
	treated_by
	treats
Disease-prevention	may_be_prevented_by
	may_prevent
Disease-diagnosis	may_be_diagnosed_by
	may_diagnose
	diagnosed_by
	diagnoses
Disease-finding	disease_excludes_finding
	disease_has_finding
	associated_etiologic_finding_of
	associated_finding_of
	disease_may_have_finding
	has_associated_etiologic_finding
	has_associated_finding
	is_finding_of_disease
	may_be_finding_of_disease
Sign or symptom	has_sign_or_symptom
	sign_or_symptom_of
	has_manifestation
causes	cause_of
Associated disease	associated_disease
	disease_has_associated_disease
	disease_may_have_associated_disease
	is_associated_disease_of
	may_be_associated_disease_of_disease
Others	induces
	evaluation_of

[4] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.

[5] A. Mnih and G. E. Hinton, "A scalable hierarchical distributed language model," in *Advances in neural information processing systems*, 2009, pp. 1081–1088.

[6] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *Journal of Machine Learning Research*, vol. 12, no. Aug, pp. 2493–2537, 2011.

[7] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng, "Improving word representations via global context and multiple word prototypes," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, 2012, pp. 873–882.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[9] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[10] Y. Goldberg and O. Levy, "word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method," *arXiv preprint arXiv:1402.3722*, 2014.

[11] X. Rong, "word2vec parameter learning explained," *arXiv preprint arXiv:1411.2738*, 2014.

[12] Y. Goldberg, "A primer on neural network models for natural language processing," *arXiv preprint arXiv:1510.00726*, 2015.

[13] M. Yu and M. Dredze, "Improving lexical embeddings with semantic knowledge." in *ACL (2)*, 2014, pp. 545–550.

[14] M. Faruqui, J. Dodge, S. K. Jauhar, C. Dyer, E. Hovy, and N. A. Smith, "Retrofitting word vectors to semantic lexicons," *arXiv preprint arXiv:1411.4166*, 2014.

[15] A. Celikyilmaz, D. Hakkani-Tur, P. Pasupat, and R. Sarikaya, "Enriching word embeddings using knowledge graph for semantic tagging in conversational dialog systems," in *AAAI Spring Symposium Series*, 2015.

[16] J. Cheng, Z. Wang, J.-R. Wen, J. Yan, and Z. Chen, "Contextual text understanding in distributional semantic space," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, 2015, pp. 133–142.

[17] W. Ling, C. Dyer, A. Black, and I. Trancoso, "Two/too simple adaptations of word2vec for syntax problems," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 1299–1304.

[18] R. Cotterell and H. Schütze, "Morphological word-embeddings," in *Proc. of NAACL*, 2015.

[19] T. Luong, R. Socher, and C. D. Manning, "Better word representations with recursive neural networks for morphology." in *CoNLL*, 2013, pp. 104–113.

[20] G. Zhou, T. He, J. Zhao, and P. Hu, "Learning continuous word embedding with metadata for question retrieval in community question answering," in *Proceedings of ACL*, 2015, pp. 250–259.

[21] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[22] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: a core of semantic knowledge," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 697–706.

[23] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. ACM, 2008, pp. 1247–1250.

[24] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *The semantic web*. Springer, 2007, pp. 722–735.

[25] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka Jr, and T. M. Mitchell, "Toward an architecture for never-ending language learning." in *AAAI*, vol. 5, 2010, p. 3.

[26] O. Bodenreider, "The unified medical language system (umls): integrating biomedical terminology," *Nucleic acids research*, vol. 32, no. suppl 1, pp. D267–D270, 2004.

[27] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*. Association for Computational Linguistics, 2009, pp. 1003–1011.

[28] M. Liu, Y. Ling, Y. An, and X. Hu, "Relation extraction from biomedical literature with minimal supervision and grouping strategy," in *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on*. IEEE, 2014, pp. 444–449.

[29] A. Bordes, S. Chopra, and J. Weston, "Question answering with subgraph embeddings," *arXiv preprint arXiv:1406.3676*, 2014.

[30] B. Tang, H. Cao, X. Wang, Q. Chen, and H. Xu, "Evaluating word representation features in biomedical named entity recognition tasks," *BioMed research international*, vol. 2014, 2014.

[31] T. Muneeb, S. K. Sahu, and A. Anand, "Evaluating distributed word representations for capturing semantics of biomedical concepts," *ACL-IJCNLP 2015*, p. 158, 2015.

[32] B. Chiu, G. Crichton, A. Korhonen, and S. Pyysalo, "How to train good word embeddings for biomedical nlp," *ACL 2016*, p. 166, 2016.

[33] B. Chiu, A. Korhonen, and S. Pyysalo, "Intrinsic evaluation of word vectors fails to predict extrinsic performance," *ACL 2016*, p. 1, 2016.

[34] P. Stenetorp, H. Soyer, S. Pyysalo, S. Ananiadou, and T. Chikayama, "Size (and domain) matters: Evaluating semantic word space representations for biomedical text," in *Proceedings of the 5th International Symposium on Semantic Mining in Biomedicine*, 2012.

[35] A. R. Aronson, "Effective mapping of biomedical text to the umls metathesaurus: the metamap program." *Proceedings of AMIA Symposium*, pp. 17–21, 2001.

[36] G. K. Savova, J. J. Masanz, P. V. Ogren, J. Zheng, S. Sohn, K. C. Kipper-Schuler, and C. G. Chute, "Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications," *Journal of the American Medical Informatics Association*, vol. 17, no. 5, pp. 507–513, 2010.

[37] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering." in *NIPS*, vol. 14, 2001, pp. 585–591.

[38] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, 2011.

[39] T. K. Landauer and S. T. Dumais, "A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge." *Psychological review*, vol. 104, no. 2, p. 211, 1997.

[40] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, "Placing search in context: The concept revisited," in *Proceedings of the 10th international conference on World Wide Web*. ACM, 2001, pp. 406–414.

[41] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Paşca, and A. Soroa, "A study on similarity and relatedness using distributional and wordnet-based approaches," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2009, pp. 19–27.

[42] H. Rubenstein and J. B. Goodenough, "Contextual correlates of synonymy," *Communications of the ACM*, vol. 8, no. 10, pp. 627–633, 1965.

[43] F. Hill, R. Reichart, and A. Korhonen, "Simlex-999: Evaluating semantic models with (genuine) similarity estimation," *Computational Linguistics*, 2016.

[44] S. Pakhomov, B. McInnes, T. Adam, Y. Liu, T. Pedersen, and G. B. Melton, "Semantic similarity and relatedness between clinical terms: an experimental study," in *AMIA annual symposium proceedings*, vol. 2010. American Medical Informatics Association, 2010, p. 572.

[45] K. Roberts, M. S. Simpson, E. Voorhees, and W. R. Hersh, "Overview of the trec 2015 clinical decision support track," in *In proceedings of TREC*, 2015.

[46] A. R. Aronson and F.-M. Lang, "An overview of metamap: historical perspective and recent advances," *Journal of the American Medical Informatics Association*, vol. 17, no. 3, pp. 229–236, 2010.

[47] S. Robertson and H. Zaragoza, *The probabilistic relevance framework: BM25 and beyond*. Now Publishers Inc, 2009.

[48] S. Balaneshin-kordan, A. Kotov, and R. Xisto, "Wsu-ir at trec 2015 clinical decision support track: Joint weighting of explicit and latent medical query concepts from diverse sources," in *Proceedings of the 2015 Text Retrieval Conference*, 2015.