RC-NET: A General Framework for Incorporating Knowledge into Word Representations

Chang Xu
College of Computer and
Control Engineering
Nankai University
Tianjin, 300071, P. R. China
changxu@nbjl.nankai.edu.cn

Yalong Bai
School of Computer Science
and Technology
Harbin Institute of Technology
Harbin, 150001, P. R. China
ylbai@mtlab.hit.edu.cn

Jiang Bian, Bin Gao
Microsoft Research
13F, Bldg 2, No. 5, Danling St
Beijing, 100080, P. R. China
{jibian,bingao}@microsoft.com

Gang Wang, Xiaoguang Liu
College of Computer and
Control Engineering
Nankai University
Tianjin, 300071, P. R. China
{liuxg,wgzwp}@nbjl.nankai.edu.cn

Tie-Yan Liu
Microsoft Research
13F, Bldg 2, No. 5, Danling St
Beijing, 100080, P. R. China
tyliu@microsoft.com

ABSTRACT

Representing words into vectors in continuous space can form up a potentially powerful basis to generate high-quality textual features for many text mining and natural language processing tasks. Some recent efforts, such as the skip-gram model, have attempted to learn word representations that can capture both syntactic and semantic information among text corpus. However, they still lack the capability of encoding the properties of words and the complex relationships among words very well, since text itself often contains incomplete and ambiguous information. Fortunately, knowledge graphs provide a golden mine for enhancing the quality of learned word representations. In particular, a knowledge graph, usually composed by entities (words, phrases, etc.), relations between entities, and some corresponding meta information, can supply invaluable relational knowledge that encodes the relationship between entities as well as categorical knowledge that encodes the attributes or properties of entities. Hence, in this paper, we introduce a novel framework called RC-NET to leverage both the relational and categorical knowledge to produce word representations of higher quality. Specifically, we build the relational knowledge and the categorical knowledge into two separate regularization functions, and combine both of them with the original objective function of the skip-gram model. By solving this combined optimization problem using back propagation neural networks, we can obtain word representations enhanced by the knowledge graph. Experiments on popular text mining and natural language processing tasks, including analogical reasoning, word similarity, and topic prediction, have all demonstrated that our model can significantly improve the quality of word representations.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning; I.5.1 [**Pattern Recognition**]: Models

General Terms

Algorithms, Experimentation

Keywords

Distributed word representations, deep learning, knowledge graph

1. INTRODUCTION

Deep learning techniques have been frequently used to solve natural language processing (NLP) tasks [8, 1, 12, 21, 22]. The main purpose of them is to learn distributed representations of words (i.e., word embedding) from text, and use them as components or the basis to generate textual features for solving NLP tasks. Recently, some efficient methods, such as the continuous bag-of-word model (CBOW) and the continuous skip-gram model [18], have been proposed to leverage the context of each word in text streams to learn word embedding, which can capture both the syntactic and the semantic information among words. The principle behind these models is that words that are syntactically or semantically similar should also have similar context words.

Although these works have demonstrated their effectiveness in a number of NLP tasks, they still suffer from some limitations. In particular, as these works learn word representations mainly based on the word co-occurrence information, the obtained word embedding cannot capture the relationship between two syntactically or semantically similar words if either of them yields very little context information. On the other hand, even enough amount of context could be noisy or biased such that they cannot reflect the inherent relationship between words and further mislead the

CIKM'14, November 3–7, 2014, Shanghai, China.
Copyright 2014 ACM 978-1-4503-2598-1/14/11 ...$15.00.
http://dx.doi.org/10.1145/2661829.2662038.

training process. To solve these problems, we propose to incorporate the information from knowledge graphs into the learning process in order to produce better word representations.

Knowledge graph is a kind of knowledge base, which has been widely used to store complex structured and unstructured knowledge. It is usually in the form of a directed or undirected graph that leverages vertices and edges to represents entities (words, phrases, etc.) and their relationships, respectively. Knowledge graphs, such as Freebase [10] and WordNet [24], have started playing important roles in many text mining and NLP applications, including expert system, question-answer system, etc. A knowledge graph commonly contains two forms of knowledge: relational knowledge and categorical knowledge. Specifically, relational knowledge (like is-a, part-of, child-of, etc.) encodes the relationship between entities so as to differentiate word pairs with analogy relationships; categorical knowledge (like gender, location, etc.) encodes the attributes and properties of entities, according to which similar words can be grouped into the meaningful categories. Both relational and categorical knowledge extracted from the knowledge graph, an example of which is shown in Figure 1, can serve as valuable external information to enhance learning word representations. Specifically, in the learning process, the relational knowledge can be leveraged to infer certain explicit connections between the embeddings of related words, and the categorical knowledge can be used to reflect coherence between the embeddings of those words with the same attributes, even if some of them yield very little context information, or biased/noisy context information.

In this paper, we propose a novel framework to take advantage of both relational and categorical knowledge to produce high-quality word representations. This framework is built upon the skip-gram model [18], in which we extend its objective function by incorporating the external knowledge as regularization functions. In particular, to leverage the relational knowledge, we define a corresponding regularization function by inheriting the similar idea from a recent study on multi-relation model [5], which characterizing the relationships between entities by interpreting them as translations in the low-dimensional embeddings of the entities. To incorporate the categorical knowledge, we define another regularization function by minimizing the weighted distance between those words with the same attributes. Then, we combine these two regularization functions with the original objective function of the skip-gram model. After solving this combined optimization problem via back propagation neural networks, we can obtain the continuous representations of words. We call the proposed framework as *RC-NET*, indicating the incorporation of both *R*elational and *C*ategorical knowledge into neural *NET*works to learn word embeddings. We have conducted empirical experiments on three popular text mining and NLP tasks, including analogical reasoning, word similarity, and topic prediction, with large-scale public datasets, and the results all demonstrate that, compared with the state-of-the-art methods, our proposed approach can significantly improve the quality of word representations by encoding both the word co-occurrence information and the external knowledge.

The rest of the paper is organized as follows. We briefly review the related work on learning word embedding via deep neural networks in Section 2. In Section 3, we describe the proposed framework to incorporate relational and categorical knowledge in learning word representations. The experimental setup and results are reported in Section 4. The paper is concluded in Section 5.

2. RELATED WORK

Building distributed word representations [14] has attracted increasing attention in the area of machine learning. Recently, to show its effectiveness in a variety of text mining and NLP tasks, a series of works applied deep learning techniques to learn high-quality word representations. For example, Collobert et al. [7, 8] proposed a neural network that can learn a unified word representations suited for several NLP tasks simultaneously. Furthermore, Mikolov et al. proposed efficient neural network models for learning word representations, i.e., word2vec [18]. This work introduced two specific models, including the continuous bag-of-words model (CBOW) and the continuous skip-gram model (skip-gram), both of which are unsupervised models learned from large-scale text corpora. Under the assumption that similar words yield similar context, these models maximize the log likelihood of each word given its context words within a sliding window. The learned word representations amazingly show that they can indicate both syntactic and semantic regularities.

Nevertheless, since most of existing works learned word representations mainly based on the word co-occurrence information, it is quite difficult to obtain high quality embeddings for those words with very little context information; on the other hand, large amount of noisy or biased context could give rise to ineffective word embeddings either. Therefore, it is necessary to introduce extra knowledge into the learning process to regularize the quality of word embedding. Unfortunately, there are very few previous studies that attempt to explore knowledge powered word embedding.

Some efforts have paid attention to learn word embedding in order to address knowledge base completion and enhancement [6, 21, 23]; however, they did not investigate the other side of the coin, i.e., leveraging knowledge to enhance word representations. Recently, there have been some early attempts on this direction. For example, Luong et al. [16] proposed a neural model to learn morphologically-aware word representations by combining recursive neural network and neural language model. In this model, they explicitly utilize the knowledge in terms of morphological structure inside a word and regard each morpheme as a basic unit. While being restricted to the morpheme-level knowledge, this attempt has not taken investigation on more important word-level knowledge, such as analogical relation between words. In contrast, we will mainly explore how to take advantage of word-level knowledge to enhance word embedding in this paper.

Most recently, Yu et al. [25] attempted to use knowledge to improve word representations. In particular, they proposed a new learning objective that incorporates both a neural language model objective and a semantic prior knowledge objective. By leveraging the knowledge in terms of semantic similarity between words during the learning process, they demonstrate that their new method can result in improvement by evaluations on three tasks: language modeling, measuring semantic similarity, and predicting human judgments. However, this model is specified on incorporating semantic knowledge and it does not explicitly distin-

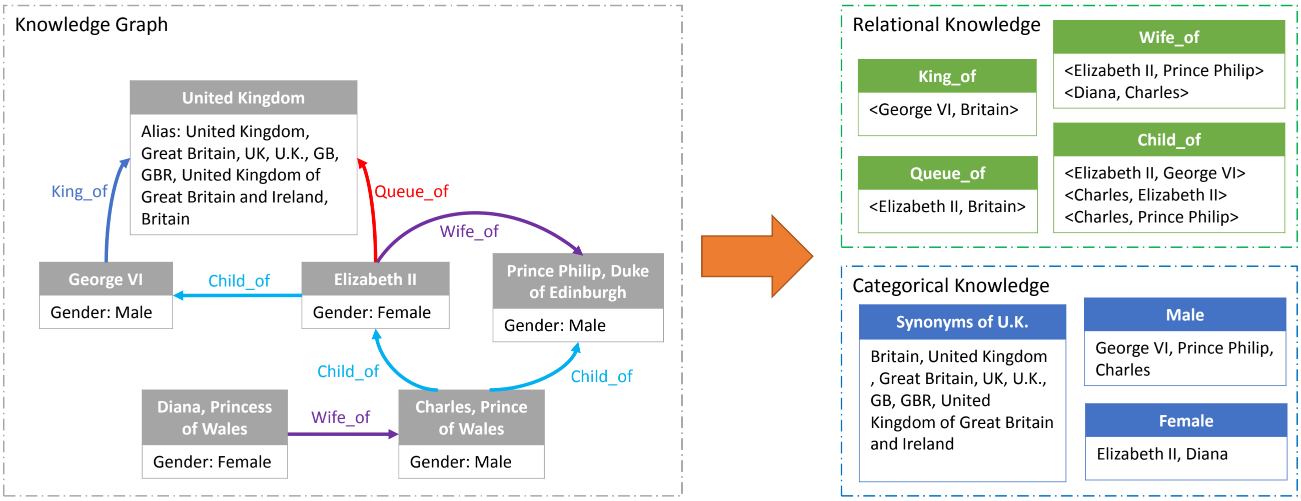


Figure 1: Knowledge graph contains two forms of knowledge: relational knowledge and categorical knowledge.

guish different kinds of relational knowledge. Bian et al. [2] recently proposed to leverage morphological, syntactic, and semantic knowledge to advance the learning of word embeddings. Particularly, they explored these types of knowledge to define new basis for word representation, provide additional input information, and serve as auxiliary supervision in the learning process. Although they did intensive empirical study, they did not make model-level innovation to leverage external knowledge to improve word representations.

In contrast to all the aforementioned works, in this paper, we present a general method to leverage various types of knowledge into learning word representations. With the target at incorporating more extensive forms of knowledge, we define a new learning objective as a combination between that of the raw text and that of external knowledge. As a result, our new model is able to learn word representations with encoding both contextual information and extra knowledge, which is much more general and flexible than previous works.

3. KNOWLEDGE POWERED WORD REPRESENTATIONS

In this section, we first introduce the continuous skip-gram model, which serves as the basis of the proposed framework. Then, we describe how we model relational knowledge and categorical knowledge as regularization functions. After that, we introduce the proposed RC-NET framework by incorporating these regularization functions into the skip-gram model to strengthen the learning of word representations. At last, we describe how we solve the optimization problem in the proposed framework.

3.1 Skip-gram

We take the continuous skip-gram model [18] as the basis of our proposed framework.[1] Skip-gram is a recently proposed algorithm for learning word representations using a neural network model, whose underlying principle lies in that similar words should have similar contexts. In the skip-gram model (see Figure 2), a sliding window is employed on the input text stream to generate the training samples. In each sliding window, the model tries to use the central word as input to predict the surrounding words. Specifically, the input word is represented in the 1-of-V format, where V is the size of the entire vocabulary of the training data and each word in the vocabulary is represented as a long vector with only one non-zero element. In the feed-forward process, the input word is first mapped into the embedding space by the weight matrix M. After that, the embedding vector is mapped back to the 1-of-V space by another weight matrix M', and the resulting vector is used to predict the surrounding words after conducting *softmax* function on it. In the back-propagation process, the prediction errors are propagated back to the network to update the two weight matrices. After the training process converges, the weight matrix M is regarded as the learned word representations.

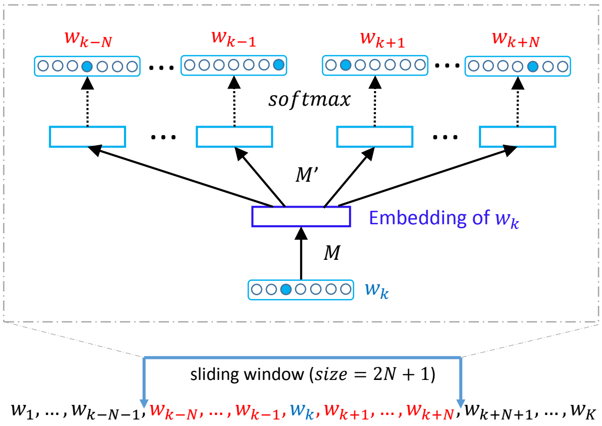

w_{k-N}
w_{k-1}
w_{k+1}
w_{k+N}
$softmax$
M'
Embedding of w_k
M
w_k
sliding window ($size = 2N + 1$)
$w_1, \dots, w_{k-N-1}, w_{k-N}, \dots, w_{k-1}, w_k, w_{k+1}, \dots, w_{k+N}, w_{k+N+1}, \dots, w_K$

Figure 2: The continuous skip-gram model.

[1] Note that although we use the skip-gram model as an example to illustrate our framework, the similar framework can be developed on the basis of any other word embedding models.

Specifically, given a sequence of training text stream $w_1, w_2, w_3, ..., w_K$, the objective of the skip-gram model is to maximize the following average log probability:

$$L = \frac{1}{K} \sum_{k=1}^{K} \sum_{-N \leq j \leq N, j \neq 0} \log p\left(w_{k+j} \mid w_k\right) \tag{1}$$

where w_k is the central word, w_{k+j} is a surrounding word, and N indicates the context window size to be $2N+1$. The conditional probability $p(w_{k+j}|w_k)$ is defined in the following *softmax* function:

$$p\left(w_{k+j} \mid w_k\right) = \frac{\exp\left({v'_{w_{k+j}}}^{T} v_{w_k}\right)}{\sum_{w=1}^{V} \exp\left({v'_{w}}^{T} v_{w_k}\right)} \tag{2}$$

where v_w and v'_w are the input and the output latent variables, i.e., the input and output representation vectors of w, and V is the vocabulary size.

To calculate the prediction errors for back propagation, we need to compute the derivative of $p(w_{k+j}|w_k)$, whose computation cost is proportional to the vocabulary size V. As V is often very large, it is difficult and sometimes impractical to directly compute the derivative. The typical method to solve this problem is noise-contrastive estimation (NCE) [13], which aims at fitting unnormalized probabilistic models. NCE can approximate the log probability of *softmax* by performing logistic regression to discriminate between the observed data and some artificially generated noise. It has been applied to the neural probabilistic language model [20] and the inverse vector log-bilinear model [19]. A simpler method to deal with the problem is negative sampling (NEG) [18], which generates l noise samples for each input word to estimate the target word, in which l is a very small number compared with V. Therefore, the training time yields linear scale to the number of noise samples and it becomes independent of the vocabulary size. Suppose the frequency of word w is $u(w)$, then the probability of sampling w is usually set to $p(w) \propto u(w)^{3/4}$ [18].

3.2 Relational Knowledge Powered Model

After briefing the skip-gram model, we introduce how we equip it with the relational knowledge. According to the left part of Figure 3, relational knowledge encodes the relationship between words. Inspired by a recent study on multi-relation model [5] that builds relationships between entities by interpreting them as translations operating on the low-dimensional representations of the entities, we propose to use a function E_r as described below to capture the relational knowledge.

Specifically, the existing relational knowledge in knowledge graphs is usually represented in the triplet (*head*, *relation*, *tail*) (denoted by $(h, r, t) \in S$, where S is the set of relational knowledge), which consists of two words $h, t \in W$ (W is the set of words) and a relationship $r \in R$ (R is the set of relationships). To learn the relation representations, we make an assumption that relationships between words can be interpreted as translation operations and they can be represented by vectors. The principle in our model is that if the relationship (h, r, t) holds, the representation of the tail word t should be close to the representation of the head word h plus the representation vector of the relationship r, i.e., $h + r$; otherwise, $h + r$ should be far away from t. Note that this model learns word representations and relation representations in the same continuous embedding space.

According to the above principle, we define E_r as a margin-based regularization function over the set of relational knowledge S.

$$E_r = \sum_{(h,r,t) \in S} \sum_{(h',r,t') \in S'_{(h,r,t)}} \left[\gamma + d(h+r,t) - d(h'+r,t')\right]_+ \tag{3}$$

In the above formulation, $[x]_+$ denotes the positive part of x, $\gamma > 0$ is a margin hyperparameter, and $d(x, y)$ is the distance measure for the words in the embedding space. For simplicity, we define $d(x, y)$ as the Euclidean distance between x and y. The set of corrupted triplets $S'_{(h,r,t)}$ is defined as:

$$S'_{(h,r,t)} = \left\{(h', r, t) \mid h' \in W\right\} \bigcup \left\{(h, r, t') \mid t' \in W\right\} \tag{4}$$

which is constructed from S by replacing either the head word or the tail word by another random selected word such that $S \bigcap S' = \emptyset$.

Note that the optimization process might trivially minimize E_r by simply increasing the norms of word representations and relation representations. To avoid this problem, we use an additional constraint on the norms, which is a commonly-used trick in the literature [5, 4, 6, 15]. However, instead of enforcing the L_2-norm of the word representations to 1, we adopt a soft norm constraint on the relation representations as below:

$$r_i = 2\sigma(x_i) - 1 \tag{5}$$

where $\sigma(\cdot)$ is the sigmoid function $\sigma(x_i) = 1/(1 + e^{-x_i})$, r_i is the i-th dimension of relation vector r, and x_i is a latent variable, which guarantees that every dimension of the relation representation vector is within the range $(-1, 1)$.

By combining the skip-gram objective function and the regularization function derived from relational knowledge, we get the following combined objective J_r that incorporates relational knowledge into the word representations learning system,

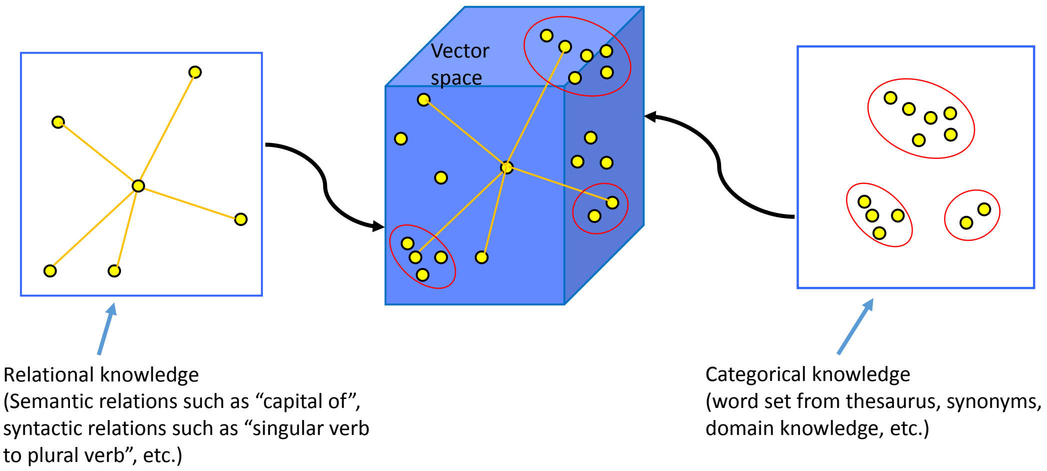
$$J_r = \alpha E_r - L \tag{6}$$

where α is the combination coefficient. Our goal is to minimize the combined objective J_r, which can be optimized using back propagation neural networks. We call this model as Relational Knowledge Powered Model, and denote it by R-NET for ease of reference.

3.3 Categorical Knowledge Powered Model

After introducing R-NET, we describe how we equip the skip-gram model with the categorical knowledge. According to the right part of Figure 3, categorical knowledge encodes the attributes or properties of words, from which we can group similar words according to their attributes. Then we may require the representations of words that belong to the same category to be close to each other.

For a specific kind of categorical knowledge, it can be represented by a similarity matrix Q, in which the element $q(w_i, w_j)$ is the similarity score between w_i and w_j. Note that many kinds of categorical knowledge can be mined from knowledge graphs, and they might vary a lot in their similarity properties. For example, in Figure 1, we can see that



Relational knowledge
(Semantic relations such as “capital of”,
syntactic relations such as “singular verb
to plural verb”, etc.)

Categorical knowledge
(word set from thesaurus, synonyms,
domain knowledge, etc.)

Figure 3: RC-NET: leveraging relational knowledge and categorical knowledge to improve the quality of word representations.

the categorical knowledge “synonyms of United Kingdom” only consists of several entities including United Kingdom, Great Britain, U.K., etc., and these words are strongly similar to each other since they are all aliases of the United Kingdom; in the same figure, we can also find that the categorical knowledge “Male” or “Female” consists of a massive number of person names, but these persons are similar only because they are all men or women, which is a very weak similarity. From the above examples, we can observe that, in most cases, categorical knowledge with smaller capacity is more likely to contain more specific information, so that we are more confident in grouping the words with that categorical knowledge close to each other. On the contrary, the categorical knowledge with larger capacity is more likely to contain more general information, so that we are less confident in grouping the corresponding words. We use this heuristic to constrain the similarity scores:

$$\sum_{j=1}^{V} s(w_i, w_j) = 1, \tag{7}$$

where if a word shares the same category with many other words, their mutual similarity scores will become small. Then, we encode the categorical knowledge using another regularization function E_c:

$$E_c = \sum_{i=1}^{V}\sum_{j=1}^{V} s(w_i, w_j) d(w_i, w_j) \tag{8}$$

where $d(w_i, w_j)$ is the distance measure for the words in the embedding space and $s(w_i, w_j)$ serves as a weighting function. Again, for simplicity, we define $d(w_i, w_j)$ as the Euclidean distance between w_i and w_j.

By combining the skip-gram objective function and the regularization function derived from the categorical knowledge, we get the following combined objective J_c that incorporates categorical knowledge into the word representations learning system,

$$J_c = \beta E_c - L \tag{9}$$

where β is the combination coefficient. Our goal is to minimize the combined objective J_c, which can be optimized using back propagation neural networks. We call this model as Categorical Knowledge Powered Model, and denote it by C-NET for ease of reference.

3.4 Joint Knowledge Powered Model

After describing the R-NET and C-NET models, it is natural to combine them into a global framework which can leverage both relational knowledge and categorical knowledge to learn word representations. Specifically, in the global framework, we want to minimize the following combined objective function:

$$J = \alpha E_r + \beta E_c - L. \tag{10}$$

We call this framework as Joint Knowledge Powered Model, and denote it by RC-NET for ease of reference.

Figure 4 shows the architecture of the proposed RC-NET framework. Compared to either of R-NET and C-NET, RC-NET shows strong superiorities. Relational knowledge mainly helps build the global structure of the learned word representations by utilizing the relationship between different words; while categorical knowledge helps improve the local structure of the learned word representations by clustering similar words together. Hence, RC-NET might yield to a structured embedding space and reduce the randomness of word representations caused by the incomplete or biased training information. Actually with the RC-NET framework, the relational knowledge and categorical knowledge can compensate each other. On one hand, sometimes the relational knowledge of some words might be absent, but we can obtain their similar words from categorical knowledge and then make inference on their relations according to the relationships of their similar words. On the other hand, sometimes the categorical knowledge of a word might be missing. However, if the word share the same relationships with a number of other words, we will be able to infer its categorical knowledge from the categories of these related words.

3.5 Optimization Procedure

In the implementation, we optimize the regularization functions derived from both the relational knowledge and the

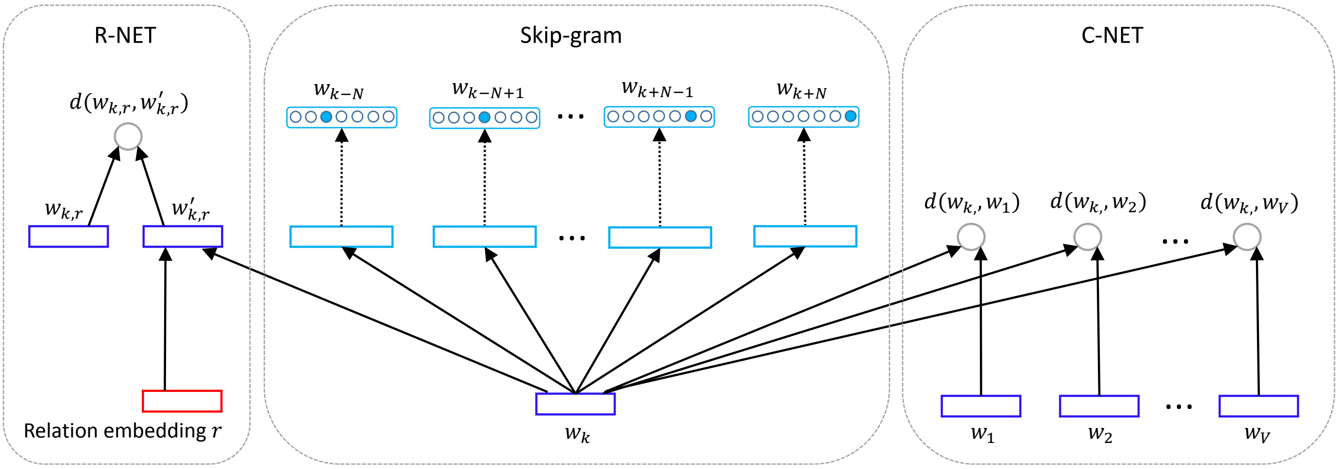


Figure 4: The architecture of RC-NET. The objective is to learn word representations and relation representations based on text stream, relational knowledge, and categorical knowledge.

categorical knowledge along with the training process of the skip-gram model. During the procedure of learning word representations from the context words in the sliding window, if the central word w_k hits the external knowledge, the corresponding optimization process of the knowledge based regularization function will be activated.

- For the branch of relational knowledge, according to the objective function (6), we minimize the Euclidean distance between the representation vector of the real tail word t and the predicted vector that is computed as $w_k + r$, and then we update the central word representation as well as the relation representation. For efficiency, we sample a subset of $S^{'}$ with a fixed size (which is also a parameter) instead of using the whole set of $S^{'}$. Note that the central word w_k may appear as the head word or the tail word in a triplet of relational knowledge.

- For the branch of categorical knowledge, we minimize the weighted Euclidean distance between the representations of the central word and that of its similar words according to the objective function (9).

The two knowledge branches and the skip-gram branch share the same word representations in the learning process. In our implementation, the optimization is conducted by stochastic gradient descent in a mini-batch mode, whose computational complexity is comparable to that of the optimization process of the skip-gram model.

4. EXPERIMENTS

In this section, we conduct experiments to examine whether incorporating relational knowledge and categorical knowledge into learning continuous word representations can significantly improve the quality of word embeddings. In particular, we compare the performance of our knowledge powered model and that of the state-of-the-art baselines by evaluating the quality of respective learned word embedding on three text mining and NLP tasks, including analogical reasoning, word similarity, and topic prediction. In the rest of this section, we first introduce the experimental setup, and then report evaluation results and further analysis on the analogical reasoning task, the word similarity task, and the topic prediction task, respectively.

4.1 Experimental Setup

4.1.1 Training Data

In our experiments, we trained word embeddings on a publicly available text corpus[2], a dataset about the first billion characters from Wikipedia. After being pre-processed by removing all the HTML meta-data and hyper-links and replacing the digit numbers into English words, the final training corpus contains totally 123.4 million words, where the number of unique words, i.e., the vocabulary size, is about 220 thousand.

4.1.2 Parameter Setting for Compared Methods

In the following experiments, we will compare four methods: **Skip-gram** (baseline), **R-NET**, **C-NET**, and **RC-NET**. To train the word embedding using these four methods, we apply the same setting for their common parameters. Specifically, the count of negative samples was set to 3; the context window size was set to 5; each model was trained through 1 epoch; the learning rate was initialized as 0.025 and was set to decrease linearly so that it approached zero at the end of training.

Moreover, the combination weights in R-NET, C-NET, and RC-NET also play a critical role in producing high-quality word embedding. Overemphasizing the weight of the original objective of Skip-gram may result in weakened influence of knowledge, while putting too large weight on knowledge powered objectives may hurt the generality of learned word embedding. According to our empirical experience, it is a better way to decide the objective combination weights of the Skip-gram model, relational knowledge, and categorical knowledge based on the scale of their respective derivatives during optimization. Specifically, it is better to set the objective weight of C-NET (β) as a smaller value

[2] http://mattmahoney.net/dc/enwik9.zip

than the objective weight of Skip-gram and R-NET since the derivative of C-NET objective usually yields a larger scale than that of Skip-gram and R-NET. Along our experiments in the following, we set $\alpha = 1$ for R-NET, $\beta = 0.001$ for C-NET, and $\alpha = 1$, $\beta = 0.0001$ for RC-NET. Note that, this parameter setting may not be optimal for different training corpus or various tasks, but the following experiments may illustrate its robustness to some extent.

4.2 Analogical Reasoning Task

4.2.1 Task Description

The analogical reasoning task was originally introduced by Mikolov *et al* [18, 17], which defines a comprehensive test set that contains five types of semantic analogies and nine types of syntactic analogies[3]. For example, to solve semantic analogies such as *Germany* : *Berlin* = *France* : *?*, we need to find a vector x such that the embedding of x, denoted as vec(x) is the closest to vec(*Berlin*) - vec(*Germany*) + vec(*France*) according to the cosine distance. This specific example is considered to have been answered correctly if x is *Paris*. Another example of syntactic analogies is *quick* : *quickly* = *slow* : *?*, the correct answer of which should be *slowly*.

In our experiments, we use an enlarged dataset called *WordRep* [11] which extends the original evaluation dataset of analogical reasoning task. In particular, this larger dataset is generated by extracting more analogy pairs from Longman dictionary[4]. Finally, we collect totally 34,773 relevant word pairs in the enlarged dataset. In our experiments, we split the whole dataset into two parts with a ratio of 4:1, in which the larger part is used for training and the smaller part for testing. To form up the testing set from the smaller part of dataset, we connect every two-word pairs from the same relation together to generate a set of four-word tuple as analogical questions. *Note that we avoid using those word pairs for training if at least one of their two words also appears in the testing set.*

4.2.2 Applied Knowledge

R-NET. For training R-NET, we directly use relation pairs and relation types as supervised information to learn representation vectors of different relations.

C-NET. For training C-NET, we extract the categorical knowledge from those relations. Specifically, given one relation, the set of head words extracted from all pairs of this relation forms up a category, and the collection of tail words forms up another category. For instance, there are 1467 "city-in-state" word pairs in the training part of the WordRep dataset. We split them into two categories: one is the collection of cities, while the other corresponds to the set of states. Each of them will be treated as a type of categorical knowledge for training C-NET.

RC-NET. Finally, we employ all the relational and categorical knowledge applied for training R-NET and C-NET in the learning process of RC-NET.

4.2.3 Experimental Results

In our experiments on the analogical reasoning task, we compared the baseline word embedding trained by Skip-gram against those trained by R-NET, C-NET, and RC-NET. The dimension of word embedding is set as 100 and 300. Table 1 illustrates the semantic, syntactic, and total accuracy by using the four methods. From this table, we can find that all of the knowledge powered models outperform the baseline skip-gram model, and RC-NET yields the largest improvements. These results can imply that the knowledge powered word embedding is of higher quality than the baseline model with no knowledge regularizations.

From Table 1, we can also observe that incorporating relational knowledge to the skip-gram model can increase the accuracy of all three sub-types of the analogical reasoning task; meanwhile, incorporating the categorical knowledge can give rise to a higher accuracy on semantic analogies but a decreasing performance on the syntactic analogies. We hypothesize the reason as, there are some syntactic relationships, such as "opposite", whose head or tail word collection do not strictly form up a word group representing a coherent category.

In order for deeper understanding on why our new methods can learn higher-quality word embedding, we take case studies on a specific syntactic relationship called "Adjective to Adverb" and a specific semantic relationship called "Male to Female". In particular, we apply the two-dimensional PCA projection on the 100-dimensional learned word embedding of randomly selected word pairs. Figure 5 reveals the RC-NET's capability of learning the representations of relational knowledge and that of constructing the distributions of words in the embedding space. In other words, from this figure, it is easy to see that, by incorporating relational knowledge, R-NET can produce word embedding such that the offset vector of any word pair in the same relationship tends to yield a common direction with similar distance, while by incorporating categorical knowledge, C-NET attempts to generate word embedding such that those words corresponding to the same topic or domain tend to be close to each other.

4.3 Word Similarity Task

4.3.1 Task Description

Another standard dataset for evaluating vector space models is the WordSim-353 dataset [9], which consists of 353 pairs of nouns. Each pair is presented without context and associated with 13 to 16 human judgments on similarity and relatedness on a scale from 0 to 10. For example, (*cup*, *drink*) received an average score of 7.25, while (*cup*, *substance*) received an average score of 1.92. To evaluate the quality of learned word embeddings, we compute Spearman's ρ correlation between the similarity scores computed based on learned word embeddings and human judgments.

Since this task expects those similar or highly-correlated words are close to each other, it could only need to incorporate the categorical knowledge extracted from similar or highly-correlated words. Thus, we only evaluate the effectiveness of C-NET for this task.

4.3.2 Applied Knowledge

To train C-NET, it is necessary to collect the categorical knowledge that can reflect the topic or concept information of words. In our experiments, we extract such knowledge for training from Freebase [3]. As a structured knowledge base, Freebase organizes words according to a couple of basic

[3] http://code.google.com/p/word2vec/source/browse/trunk/questions-words.txt

[4] http://www.longmandictionariesonline.com/

Table 1: Performance of using relational knowledge and categorical knowledge on the analogical reasoning task based on our proposed models.

Model	Vector Dimensionality	Accuracy[%]		
		Semantic	Syntactic	Total
Skip-gram	100	25.06	36.49	31.30
	300	28.76	40.31	35.07
R-NET	100	26.91	39.37	33.56
	300	32.64	43.46	38.55
C-NET	100	29.67	36.12	33.19
	300	**37.07**	40.06	39.00
RC-NET	100	**32.02**	**43.92**	**38.52**
	300	34.36	**44.42**	**39.85**

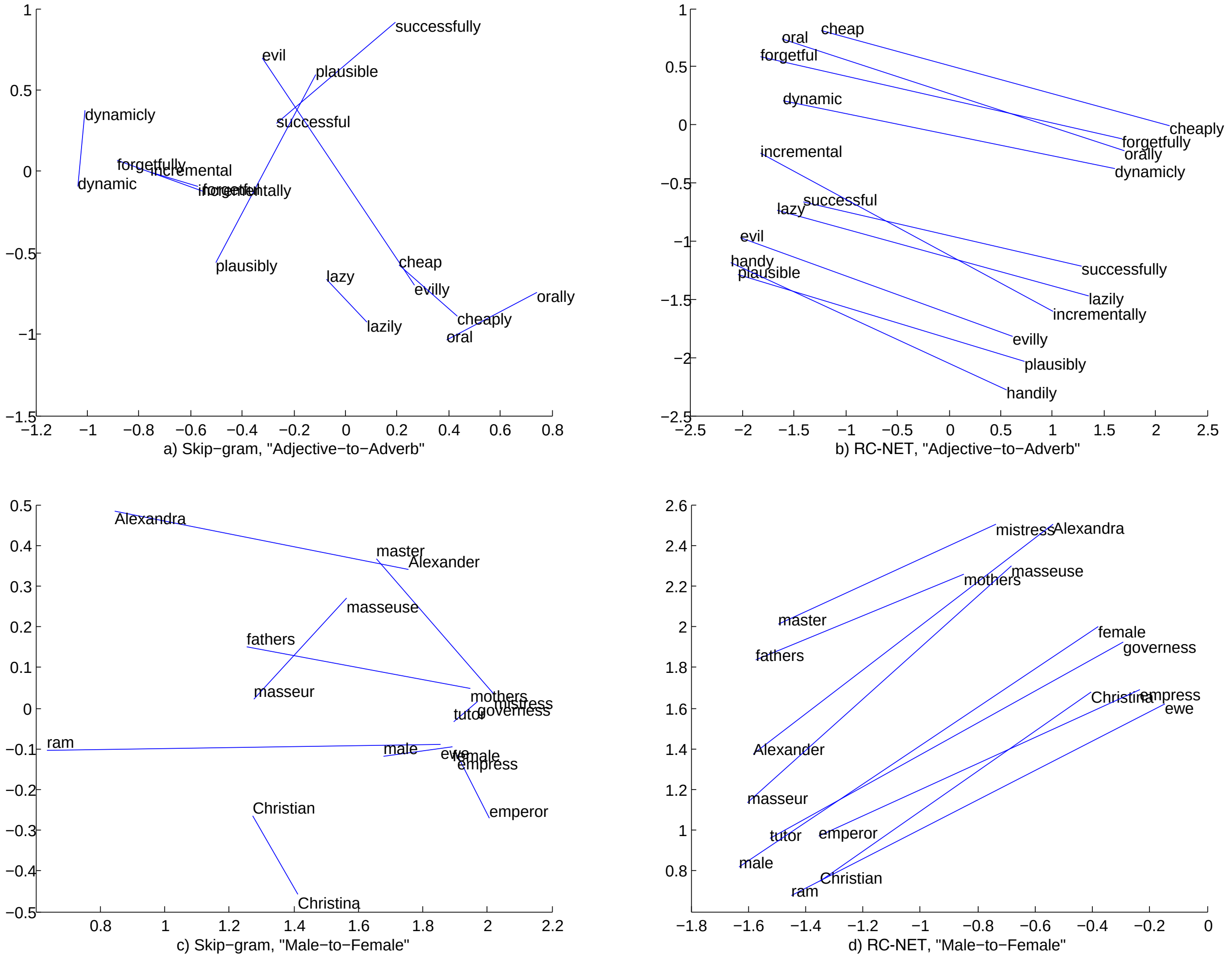


Figure 5: Two-dimensional PCA projection of 100-dimensional Skip-gram vectors and our proposed RC-NET word vectors of syntactic relation "Adjective to Adverb" and semantic relation "Male to Female". All of these word pairs were chosen randomly.

relations. Among them, we take advantage of the "type of" relation to generate the categorical knowledge since this type of relation can naturally reflect the correlation between entities (words) with topics.

While Freebase contains many domain-specific words, such as professional terminologies and names (person, location, business), these words are so rare in the general training corpus that they yield quite limited contribution to improve word embedding quality. Therefore, to address this problem, we only collect the categorical knowledge related to a pre-defined vocabulary, which only contains common nouns in the Longman Dictionary and filters our all multi-word

Table 2: Results obtained by the different methods on the word similarity task.

Methods	Vector Dimensionality	Spearman's ρ correlation
Skip-gram	100	0.652
	300	0.678
C-NET	100	**0.661**
	300	**0.683**

phrases and non-alphabetic characters. Finally, we select the top 10 human rated topic sets, including astronomy, biology, boats, chemistry, computer, fashion, food, geology, interests, and language, as the categorical knowledge for training, the vocabulary size of which is 3,174.

4.3.3 Experimental Results

Table 2 compares the performance of C-NET against Skip-gram on the word similarity task. From this table, we can find that C-NET can achieve better performance than Skip-gram on this task no matter 100 or 300 vector dimension is applied. These results indicate that C-NET can more effectively let those words similar in terms of topic or concept be close to each other in the obtained representation space, as it incorporates the categorical knowledge extracted from Freebase explicitly into the learning process so as to encode the similarity and correlation between words into the word representations.

4.4 Topic Prediction Task

4.4.1 Task Description

In many text mining and NLP applications, it is important to identify the topic of any given word since it can provide useful semantic information. For instance, both the word "star" and "earth" correspond to the topic of "astronomy", and both "cell" and "neuron" belong to the topic of "biology". In the rest of this section, we evaluate word embedding via the topic prediction task, whose goal is to find the most related topic word for a given word.

Our proposed methods, especially R-NET and RC-NET can be naturally applied to solve this task. In particular, since R-NET and RC-NET can learn the relation embedding beyond word embedding, given a word h and the *topic relation* embeddings r, we can predict the topic word t as the one that has the shortest Euclidean distance to $h + r$ over the whole vocabulary. Although Skip-gram and C-NET do not explicitly produce the relation embedding in the training process, for a specific relation r, we are able to compute the average offset vector $t - h$ for any word pair $\langle h, t\rangle$ belonging to this relation as the embedding of r. Then, we can follow the same way of R-NET and RC-NET to solve the topic prediction task.

4.4.2 Applied Knowledge

For training C-NET for this task, we leverage the same categorical knowledge used for the word similarity task, as described in Section 4.3.2. To obtain relational knowledge for training R-NET and RC-NET, we simply transform the dataset about categorical knowledge into a new format to represent relational knowledge. Specifically, the relational knowledge is in the triple format (h, r, t), where h is a specific word under a certain topic, r is the corresponding topic relation, and t is the name of topic or concept.

Table 3: Results obtained by the comparing methods on the topic prediction task.

Relation	Error Rate[%]			
	Skip-gram	C-NET	R-NET	RC-NET
astronomy	2.00	2.00	8.00	2.00
biology	6.29	4.91	4.91	4.32
boats	11.76	5.88	5.88	7.84
chemistry	6.67	5.71	9.52	15.24
computer	19.54	8.62	6.32	4.02
fashion	22.08	24.68	22.08	22.08
food	17.98	13.60	11.40	7.89
geology	0.00	0.00	11.54	7.69
interests	6.80	8.16	6.12	6.12
language	33.33	33.33	18.52	7.41
Total	12.21	9.37	8.57	**7.15**

4.4.3 Experimental Results

In the following experiments, we split the generated knowledge data into training set and testing set by the ratio of 1:1 for each relation. Note that there is no overlap between the training set and the testing set in the vocabulary except for the topic word t. The dimension of word representations is set as 100.

Table 3 reports the error rates of different word embedding models on the topic prediction task. From this table, we can see that knowledge powered models can achieve lower error rates than Skip-gram on most of the relations. Furthermore, RC-NET can reach better performance than R-NET and C-NET, which indicates that both relational and categorical knowledge are important for predicting the topic for the word.

From the table, we also observe that there are some relations, where knowledge powered models do not yield better performance. Our further analysis reveals that these relations can be classified into two types. One type includes relations that have inadequate training pairs such that the relation embedding cannot be trained sufficiently. For example, it is quite difficult to train high quality embedding for the relation "astronomy" and "geology" since they merely have 25 and 13 pairs for training, respectively. The other type contains relations which have so many rare words that they yield less chance to be trained either. For example, as there are a lot of uncommon words in the relation "chemistry", it is not easy to collect enough training samples for this relation.

5. CONCLUSIONS AND FUTURE WORK

Learning high-quality word embedding is quite valuable for many text mining and NLP tasks. To address the limitation of the state-of-the-art methods in terms of their incapability of encoding the properties of words and the complex relationships among words very well, this paper proposes to incorporate knowledge graphs into the learning process since it contains invaluable relational knowledge that encodes the relationship between entities as well as categorical knowledge that encodes the attributes or properties of entities. In this paper, we introduce a new knowledge powered method, called RC-NET, to leverage both the relational and categorical knowledge to obtain word representations. Experiments on three popular text mining and NLP tasks have illustrated that the knowledge powered method can significantly improve the quality of word representations.

For the future work, we will explore how to incorporate more types of knowledge, such as the morphological knowledge of words, into the learning process to obtain more powerful word representations. Meanwhile, we will study how to define more general regularization functions to represent the effect of various types of knowledge.

6. REFERENCES

[1] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A neural probabilistic language model. In *The Journal of Machine Learning Research*, pages 3:1137–1155, 2003.

[2] J. Bian, B. Gao, and T.-Y. Liu. Knowledge-powered deep learning for word embedding. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2014.

[3] 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*, pages 1247–1250. ACM, 2008.

[4] A. Bordes, X. Glorot, J. Weston, and Y. Bengio. A semantic matching energy function for learning with multi-relational data. *Machine Learning*, 94(2):233–259, 2014.

[5] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, pages 2787–2795, 2013.

[6] A. Bordes, J. Weston, R. Collobert, Y. Bengio, et al. Learning structured embeddings of knowledge bases. In *AAAI*, 2011.

[7] 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*, pages 160–167. ACM, 2008.

[8] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537, 2011.

[9] 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*, pages 406–414. ACM, 2001.

[10] Freebase. http://www.freebase.com.

[11] B. Gao, J. Bian, and T.-Y. Liu. Wordrep: A benchmark for research on learning word representations. In *ICML 2014 Workshop on Knowledge-Powered Deep Learning for Text Mining*, 2014.

[12] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *In Proceedings of the Twenty-eight International Conference on Machine Learning, ICML*, 2011.

[13] M. U. Gutmann and A. Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *The Journal of Machine Learning Research*, 13:307–361, 2012.

[14] G. E. Hinton. Distributed representations. 1984.

[15] R. Jenatton, N. Le Roux, A. Bordes, G. Obozinski, et al. A latent factor model for highly multi-relational data. In *NIPS*, pages 3176–3184, 2012.

[16] M.-T. Luong, R. Socher, and C. D. Manning. Better word representations with recursive neural networks for morphology. *CoNLL-2013*, 104, 2013.

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

[18] 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*, pages 3111–3119, 2013.

[19] A. Mnih and K. Kavukcuoglu. Learning word embeddings efficiently with noise-contrastive estimation. In *Advances in Neural Information Processing Systems*, pages 2265–2273, 2013.

[20] A. Mnih and Y. W. Teh. A fast and simple algorithm for training neural probabilistic language models. *arXiv preprint arXiv:1206.6426*, 2012.

[21] R. Socher, D. Chen, C. D. Manning, and A. Ng. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems*, pages 926–934, 2013.

[22] G. Tur, L. Deng, D. Hakkani-Tur, and X. He. Towards deeper understanding: Deep convex networks for semantic utterance classification. In *ICASSP*, pages 5045–5048, 2012.

[23] J. Weston, A. Bordes, O. Yakhnenko, and N. Usunier. Connecting language and knowledge bases with embedding models for relation extraction. *arXiv preprint arXiv:1307.7973*, 2013.

[24] WordNet. “about wordnet”, princeton university. http://wordnet.princeton.edu. 2010.

[25] M. Yu and M. Dredze. Improving lexical embeddings with semantic knowledge. In *Association for Computational Linguistics (ACL)*, 2014.