# Social Relation Extraction with Improved Distant Supervised and Word Embedding Features

Jinwen Liu, Weikang Rui, Liping Zhang, Yawei Jia
School of Computer Science and Technology
University of Science and Technology of China
Hefei, 230022, China
e-mail: {jwliu1, jasonrui, sa614428,ywjia}@mail.ustc.edu.cn

***Abstract*—With the rapid development of Internet, how to extract personal relations from Internet has become an important research topic in information extraction. However, current relation extraction researches mainly focus on the processing of English language, the researches focus on Chinese are less. At the same time, there are two main problems in current personal relation extraction approaches: 1) it is difficult to get a large amount of high quality training data without manually label effort; 2) the performance of personal relation extraction from Chinese is unsatisfactory. To solve the first problem, we propose an improved distant supervision method which is applied to Chinese language and can label large-scale of high quality training data automatically. To solve the second problem, we extract three features based on word embedding and combine them with the basic features. In the experiment, the improved distant supervision methods improve the quality of training data significantly. And the word embedding features improve the average F1 score by 4% than the basic features.**

***Keywords-Personal relation extraction; Word Embedding; Distant supervision; Social network; Natural language processing***

## I. Introduction

The rapid development of the Internet brings users a flood of information resources which contain a great deal of information on the relations of person. Large-scale of personal relations are critical for the construction of knowledge base such as Freebase [1], YAGO [2] and the natural language processing tasks such as Question Answering and Knowledge Graph Construction. In this paper, we focus on the problem of extracting personal relations from web.

The approaches based on machine learning have been developed to deal with relation extraction. Supervised approaches need to label large-scale data; distant supervision relation extraction settles this problem by aligning entities in text with those in knowledge base. However, distant supervision relation extraction has been widely studied in English, while the research progress in Chinese relation extraction is quite limited due to the characteristic of Chinese language. There were many researches that use distant supervision to get training data for relation extraction, but they haven't solved the problem of wrongly labeled positive examples which was caused by the naive assumption of distant supervision. In this paper, we improve the assumption in distant supervision and propose an approach which selects the candidate positive examples by a scoring function and expand the training data by relation feature words.

Recently, word embedding has been used in different NLP tasks such as named entity recognition or parser [3], [4]. Compared with previous researches, our method focuses on personal relation extraction from Chinese free text. We suppose that combining the word embedding features with basic features can improve the performance of relation classification. In general, to classify relations between pairs of persons, the most important features come from the pairs themselves, the words between and around the pairs. So we construct the entity pair feature and Bag-of-Words feature calculated by word embedding vectors. And the experimental results show that our new features improve the performance of the relation classification significantly.

The remaining of this paper is organized as follows. We first introduce the related work in Section Ⅱ, and then we introduce the proposed relation extraction model in Section Ⅲ. In section Ⅳ, we present our experimental results and analysis. The last section is the summary of this paper and the work in the future.

## II. Related Work

Relation extraction aims to identify the semantic relation between entities from natural language text. An important problem for relation extraction approach is how to label training and test data for learning classifiers. Distant supervision for relation extraction was proposed by Mintz et al. [5], which used background information from the existing structured knowledge Freebase to label sentences and created training data for relation classifiers. Unlike supervised systems, distant supervision does not require manual effort to label data and can be applied to large corpora.

In recent years, more and more researchers pay attention to distant supervised methods [6], [7] for its effective use of available databases. The assumption of distant supervision is that any sentence contains a pair of entities ($e_1$ and $e_2$) that occur in a known relation $r$, is likely to express that relation $r(e_1, e_2)$ and thus forms a positive training example of $r$. But the assumption is limited and may cause inherent errors in the process of generating training data [8].Riedel et al. [9], Surdeanu et al. [10] and Hoffmann et al. [11] loosed the distant supervision assumption with multi-instance learning algorithms. They assumed that at-least-one mention of pair ($e_1$, $e_2$) in all mentions actually expresses the relation. However, the relaxation is equivalent to the distant supervision assumption when a labeled pair of entities is mentioned only one time in a target corpus. How to reduce the wrongly labeled positive examples in distant supervision is one of the aims in our research.

Various models for learning word embedding have been proposed, including neural language models [12] and spectral models [13]. More recently, Mikolov et al. [14] proposed two log-linear models to efficiently induce word embedding. Word embedding features have been popular as an alternative to hand-crafted features. For information extraction, Nguyen and Grishman [15] employed word embedding for domain adaptation of relation extraction. Weston et al. [16] proposed a novel approach for relation extraction from free text which was trained to use information from the text and from existing knowledge.

## III. Relation Extraction Model

### A. *Workflow of Relation Extrcation*

The outline of the proposed Chinese personal social relation extraction model is shown in the Fig. 1.

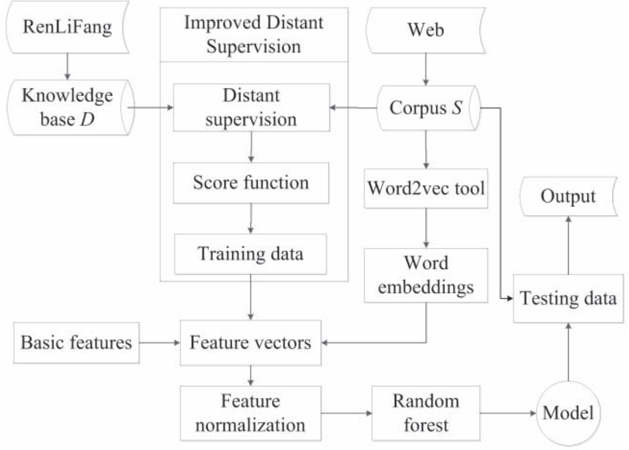

Figure 1. Outline of the proposed relation extraction model

The construction of the model can be divided into four parts in the following:

- The knowledge base $D$ is constructed from the data in ‘RenLiFang’ and the Corpus $S$ contains the sentences crawled from the web. Improved distant supervision method is used to combine these two parts of data can obtain our training data.
- Basic features and word embedding features are extracted from the training and testing data. The testing data which contain candidate tuples are extracted from the corpus $S$ and transformed into feature vectors.
- The Random Forest classifiers for each relation based on the feature system are trained.
- The classifiers are used to perform the predictions on the testing data, the output of the classifiers are the prediction results of the testing data.

The above parts will be described in detail in the following four sections.

### B. *Corpus Pre-Processing*

We construct the knowledge base $D$ by the ‘RenLiFang’ data. The data from ‘RenLiFang’ contains abundant semi-structured personal entries. From a single entry of person $p_i$, we can get following three parts of information.

The first part is the social relationships of the person $p_i$. From the raw data in ‘RenLiFang’, we extract the tuples such as tuple $t = (p_i, r, p_j)$, where $p_i$ is the person related with $p_j$ and $r$ is the relation word between them. Then the tuples are inserted into knowledge base $D$. After the data processing, 2269 kinds of relations and 1.95 million personal relation pairs are obtained. Although the relation words are abundant, many relation words express the similar meaning. So we perform the relation generalization on these relation words and transform the words with similar or same meaning into one canonical relation type. Finally, we obtain 256 types of relation and 10 highest frequency relation types.

The second part is the relation feature words. In the process of relation generalization, the fine-grained relation words compose the relation feature words set for each relation type. For example, “second daughter” can be generalized into “daughter” relation and “second daughter” is the relation feature word of “daughter” relation.

The third part of data extracted from person $p_i$ attribute is the gender attributes. The amount of person gender attribute data is 113,470 in the knowledge base $D$.

The raw data of the corpus $S$ is collected from the news context of web pages. The corpus is segmented into sentences, and then word segmentation, part-of-speech tagging and person name recognition are performed on the sentences.

### C. *Improved Distant Supervised Learning*

The relation feature words for each relation can be obtained from the knowledge base $D$, which are generated in the process of relation generalization.

After we obtained the relation knowledge base $D$ and corpus $S$, we use the seed tuples in $D$ to match the sentences in corpus $S$ to get candidate training data. If the relation word or at least one relation feature words and two person names in a tuple all exit in the sentence $s$, then the sentence $s$ is labeled with the tuple as the candidate of training data.

We propose a scoring method performed on the candidate training data. Suppose that the candidate sentence $s$ contains the tuple $t = (p_i, r, p_j)$, and $feature\_words[r] = \{r_1, r_2, \ldots, r_n\}$ are the relational feature words of relation $r$. The scoring function of sentence $s$ with tuple t is *score(s,t)*:

$$score(s,t) = \frac{w_1 * N_1 - w_2 * N_2 - w_3 * N_3 - w_4 * S_1}{Length(s)} \quad (1)$$

where $N_1$ is the number of words in *feature_words* [$r$] that appear in sentence $s$; $N_2$ is the number of person names that appear in $s$ except $p_i$ and $p_j$; $N_3$ is the number of punctuation between $p_i$ and $p_j$; $S_1$ is the shortest distance between $p_i$ and $p_j$ in $s$. $w_1$, $w_2$, $w_3$, $w_4$ are the weights of the variables in the function. We adjust the weights according to the results of obtained training data and select the sentences with highest scores as the positive examples for relation $r$.

We acquire negative cases for relation $r$ by selecting from some relation seeds of other types of relation $r^*$ and transform the tuple ($p_i$, $r^*$, $p_j$) into ($p_i$, $r$, $p_j$). Then match the corpus $S$ to acquire the negative examples. For each kind of relation, we keep the number of the positive cases and negative almost same, which is beneficial for the relation classification performance.

### D. Basic Features

We train a classifier for each relation type and transform the problem of person relation extraction to the classification problem. For a relation tuple t ($p_i$, $r$, $p_j$), which expresses that relation $r$ between person $p_i$ and person $p_j$, the basic features we extract from sentence $s$ are:

*1) Lexical Features:* These features mainly use the position information of relation word and person pairs in the sentence. Suppose the sentence s contains the relation tuple t, the lexical features are:

- The position of the relation $r$ in $s$ relative to $p_i$ , $p_j$
- The distance between $p_i$ or $p_j$ and $r$
- The number of verbs, nouns between $p_i$ or $p_j$ and $r$
- The number of relation feature words between $p_i$ or $p_j$ and $r$
- The number of person names and personal pronouns between $p_i$ or $p_j$ and $r$

*2) Syntactic Features:* Syntactic analysis is conducted on the sentence s and syntax parser tree is obtained. From the syntax tree, we can get the least common ancestor of $p_i$, $p_j$. The syntactic features are:

- The number of clauses between $p_i$ or $p_j$ and $r$
- The number of phases between $p_i$ or $p_j$ and $r$
- The part-of-speech tag of the least common ancestor of $p_i$ and $p_j$
- The distance between least common ancestors and $p_i$ or $p_j$

*3) Context Feature:* The context of the person position in the sentence is important for the relation identify. We set the context window as 2 words, which is shown in the Eq. 2. The part-of-speech of the words that before and after person name $p_i$ and $p_j$ in the Context_window are inserted into our basic features.

$$Context\_window=\{\ldots,w_{i-1},w_{i-2},p_i,w_{i+1},w_{i+2},\ldots\} \qquad (2)$$

### E. Word Embedding Features

Basic features of the sentences can only obtain the local information from the sentence, which is limited to express the sentences' meaning. Word embedding via deep learning technology can reflect the semantic correlation of words. So we introduce the word embedding features into our feature system.

Mikolov [14] proposed two log-linear models, namely the *Skip-gram* and *CBOW* model. Their experiment results had shown that the *Skip-gram* model performs better in identifying semantic relationship among words. Therefore, we use the *Skip-gram* model for training word embedding model in this study. We construct three features based on word embedding, which take advantage of different levels of information of the sentences, respectively.

*1) Entity Pair Feature:* We suppose that given the type of relation, for a pair of person $p_i$, $p_j$ that satisfies the relation, their semantic similarity calculated by embedding vectors may be similar with other pairs that with the same relation. We calculate the Euclidean distance between the two person's embedding vectors and set the distance as one of our new features. The vector of $p_i$ is $V_i = \{a_1,a_2,\ldots, a_n\}$, and the vector of $p_j$ is $V_j=\{b_1,b_2,\ldots, b_n\}$, the Euclidean distance between the two words can be calculated in the Eq. (3):

$$Dis\tan ce(V_i,V_j)=\sqrt{\sum_{i=1}^{n}(a_i-b_i)^2} \qquad (3)$$

*2) Bag-of-Words Feature*: Inspired by the Bag-of-Words model, we build a word bag of relation feature words for each type of relations, and then calculate the correlations between the person pairs and the word bags .We define this Bag-of-words correlation as *bagC*, calculated in Eq. (4), where n is the number of relation feature words, i = {1,…, n} $w_i$ is one of the relation feature word in the *bagC*:

$$bagC\ (p_A,p_B)=\frac{1}{n}(\sum_{i=1}^{n}wordsim(p_A,w_i)+\sum_{i=1}^{n}wordsim(p_B,w_i)) \qquad (4)$$

$p_A$ is person A, $p_B$ is person B, *wordsim*($w_1$,$w_2$) is the Euclidean distance between the embedding vectors of $w_1$ and $w_2$.

*3) Word Similarity Feature:* For a sentence $s$ and relation tuple $t$, the sum of semantic similarity between the relation word $r$ and other noun or verb $w_i$ in the sentence is calculated. Considering the influence of *Len*($s$), which is the length of sentence $s$, the sum of semantic similarity is divided by the *Len*($s$) as *Avg_Similarity*:

$$Avg\_Similarity=\frac{\sum_{i=1}^{n}Dis\tan ce(vec(w_i),vec(r))}{Len(s)} \qquad (5)$$

where n is the number of nouns and verbs in the sentence $s$, and the result *Avg_Similarity* is inserted into the feature vector as another feature.

## IV. Experiments

### A. Datasets

As the structured data, our experimental data gains from the website: Microsoft 'RenLiFang', from which we obtained 246,309 person entries. Then we can obtain the social relationships of the person and the person's attribute information, such as gender, birth time.

The unstructured data come from two sources: 1) we downloaded *SogouCA* corpus published by *Sougou Lab* which contains the whole web news context data from June to July in 2012; 2) we crawled news pages published from 2005 to 2014 on the large Chinese websites.

The Chinese segmentation is provided by the Chinese lexicon analysis system *ICTCLAS* and we conducted the syntactic analysis by the open-source Chinese language processing *LTP*.

We performed the word segmentation on the news data and train the word embedding with the Google's *word2vec* tool. We used the *skip-gram* model and set the size of vector dimension as 200, the size of training windows as 5. At last, we obtained the embedding vectors containing 897,304 words.

### B. Evaluation Measure

We used the Random Forest in *Scikit-learn* machine learning toolkit as the classifier. In the following experiments, we selected 10 common social person relations as the experiment relations. For each relation type in our experiment, we trained a classifier and performed the 10-fold cross validation 15 runs to calculate the Precision, Recall and F1 measure. The presented results are the average of the results across all runs.

### C. Effect of Improved Distant Supervised Learning

In this section, we set the parameters in the scoring function as $w_1$=100, $w_2$=20, $w_3$=10, $w_4$=3, with which our methods can get the best performance among the 20 sets parameters. Then we selected the sentences with the scores above the average score as the positive examples. We compared the proportion of the true positive examples by improved distant supervised and the traditional distant supervision, the result is shown in the Table I.

TABLE I. PROPORTION OF TRUE POSITIVE EXAMPLES IN TRAINING DATA

| Relations | DS (%) | Improved DS (%) |
|---|---|---|
| Son | 66 | 95 |
| Sibling | 67 | 95 |
| spouse | 65 | 94 |
| Father | 65 | 96 |
| Friend | 58 | 95 |
| Cooperate | 67 | 97 |
| Lover | 60 | 93 |
| Mother | 65 | 94 |
| Daughter | 63 | 93 |
| Teacher | 70 | 97 |

We can see that the improved distant supervision improves the quality of training data for every relation we studied significantly. The average true positive example proportion in traditional distant supervision is about 64.6%, while the average proportion is almost 95% with our method, which solves the problem caused by the inaccurate assumption in traditional distant supervision approaches. With the improved distant supervised methods, we obtained large-scale of positive and negative examples for each relation in training data.

### D. Performance of New Features

*1) Comparison of Different Classifiers:* We selected three common classifiers in machine learning tasks and tested their performances for our person relation extraction task. The three classifiers were Logistic Regression, Random Forest, SVM. We performed the parameters selecting for each classifier and these classifiers were trained with basic features. In the Fig. 2, the relation 1 to 10 represent the 10 relations we studied and we can see that the Random Forest model performs the best on F1 values over all relations. So we choose the Random Forest as our classifier in the following experiments.

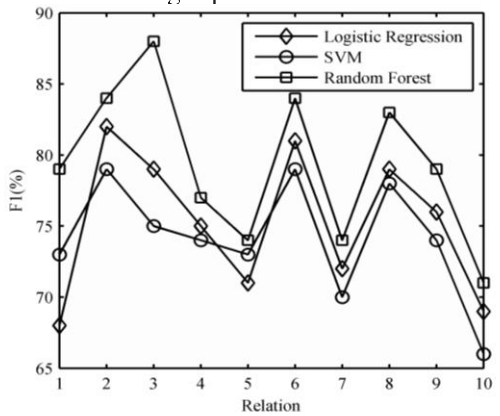

Figure 2. F1 values of different classifiers with basic features

*2) Performance of Word Embedding Feature:* We conducted several experiments among the features extracted with embedding vectors and select the combination of lexical, syntactic and context feature as the basic features in this experiment. We used $F_b$ represents the basic features, $F_p$ represents person pair feature, $F_w$ represents the Bag-of-Words correlation feature, $F_s$ represents the word similarity feature and F* represents the combination of the three features. We combined these features with $F_b$ respectively and compared the F1 values of classifiers with these features over all relations we studied, the result is shown in the Table II.

TABLE II F1 VALUES (%) OF MODELS WITH DIFFERENT FEATURES

| Relations | $F_b$ | $\cup F_p$ | $\cup F_w$ | $\cup F_s$ | $\cup F_p \cup F_w$ | $\cup F^*$ |
|---|---|---|---|---|---|---|
| Son | 76.7 | 78.1 | 77.9 | 79.1 | 79.3 | 79.8 |
| Sibling | 81.2 | 83.4 | 82.7 | 83.5 | 83.8 | 85.3 |
| Spouse | 86.6 | 88.0 | 87.8 | 88.1 | 88.3 | 89.2 |
| Father | 74.7 | 76.5 | 76.2 | 77.2 | 77.3 | 77.6 |
| Friend | 69.3 | 72.1 | 73.3 | 71.4 | 74.3 | 75.3 |
| Cooperate | 79.8 | 81.2 | 81.9 | 80.3 | 82.9 | 83.3 |
| Lover | 72.4 | 74.0 | 74.1 | 73.4 | 76.4 | 77.0 |
| Mother | 82.4 | 84.8 | 84.7 | 85.0 | 85.3 | 85.6 |
| Daughter | 77.7 | 79.7 | 79.9 | 79.8 | 80.2 | 80.7 |
| Teacher | 68.9 | 71.0 | 70.2 | 70.3 | 71.3 | 72.4 |
| Average | 77.0 | 78.9 | 78.9 | 78.8 | 79.4 | 80.6 |

We can see that the F1 value of $F_b \cup F^*$ feature performs best and $F_b$ performs the worst over all relations in the Table Ⅱ . The highest promotion is "friend" relation, which improves 6%, and the lowest promotion is "spouse" relation, which improves 2.6%. We also calculate the average F1 value for all relations, and the average promotion is 3.6%. The data in the Table Ⅱ shows that each feature we calculate based on word embedding can improve the F1 value, while the combination of them performs the best. We can improve the performance of the relation extraction system through combining the word embedding features with basic features.

*3) Performance of Gender Attribute Feature:* Some kinds of relation have obvious person gender attribute, such as "wife" and "husband". Person gender attribute can promote our classifiers' performance. We suppose "0" represents "male" and "1" represents "female", the gender information of the person pairs can be added into feature vector. We calculated the average performance of basic features, word embedding features and gender attribute features over the ten relations. $F_a$ feature represents person gender attribute. The results is shown in the Table Ⅲ.

TABLE III. AVERAGE RESULTS OVER ALL RELATIONS

| Features | Precision (%) | Recall (%) | F1 (%) |
|---|---|---|---|
| $F_b$ | 79.8 | 74.4 | 77.0 |
| $F_b \cup F_p$ | 82.2 | 75.9 | 78.9 |
| $F_b \cup F_w$ | 81.7 | 76.3 | 78.9 |
| $F_b \cup F_s$ | 81.3 | 76.4 | 78.8 |
| $F_b \cup F_p \cup F_w$ | 81.8 | 77.1 | 79.4 |
| $F_b \cup F^*$ | 82.6 | 78.6 | 80.6 |
| $F_b \cup F^* \cup F_a$ | 83.3 | 78.9 | 81.0 |

From the Table III, the feature $F_b \cup F^* \cup F_a$ improves average F1 value by 0.4% than the $F_b \cup F^*$ feature. So we can conclude that the word embedding features and the person gender attribute feature are effective for improving the performance of personal relation extraction in Chinese free text.

## V. CONCLUSION

In this paper, we proposed a new personal social relation extraction method. This method constructed a large-scale Chinese personal relation knowledge base with the online source, and obtained high quality training data by the improved distant supervision. We selected ten most frequency personal relations from the knowledge base and combined word embedding features, person gender feature with basic features to construct the classifiers for these relations.

The first part of experiment result shows that the positive proportion in training data with improved distant supervision method is much high than the traditional distant supervision. We can conclude that the methods proposed in this paper are efficient for the generation of training data without manual effort. The second part of the experiment result shows that the features based on word embedding and gender attribute improve the average F1 score over ten relations by 4% than basic features. So our personal relation extraction system based on improved distant supervision and word embedding features have the practical application value.

In future work, we want to improve word embedding learning under the guidance of entity relations. By including the entity relation constraints while training word embeddings, we expect to improve the embeddings such that they become more suitable for the relation extraction.

## REFERENCES

[1] 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.

[2] M. Fabian, K. Gjergji, and W. Gerhard, "Yago: A core of semantic knowledge unifying wordnet and wikipedia," in 16th International World Wide Web Conference, WWW, 2007, pp. 697–706.

[3] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning," in Proceedings of the 48th annual meeting of the association for computational linguistics (ACL), 2010, pp. 384–394.

[4] R. Socher, J. Bauer, C. D. Manning, and A. Y. Ng, "Parsing with compositional vector grammars," in *Proceedings of the ACL conference*. Citeseer, 2013.

[5] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proceedings of ACLIJCNLP*, vol. 2, 2009, pp. 1003–1011.

[6] G. Angeli, J. Tibshirani, J. Y. Wu, and C. D. Manning, "Combining distant and partial supervision for relation extraction," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014.

[7] M. Pershina, B. Min, W. Xu, and R. Grishman, "Infusion of labeled data into distant supervision for relation extraction," in *Proceedings of ACL*, 2014.

[8] B. Roth, T. Barth, M. Wiegand, and D. Klakow, "A survey of noise reduction methods for distant supervision," in *Proceedings of Conference on Information and Knowledge Management (CIKM-AKBC)*, 2013, pp.73–78.

[9] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD). Springer, 2010, pp. 148–163.

[10] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning, "Multi-instance multi-label learning for relation extraction," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2012, pp. 455–465.

[11] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld, "Knowledge-based weak supervision for information extraction of overlapping relations," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics(ACL)*, 2011, pp. 541–550.

[12] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *The Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003.

[13] P. Dhillon, D. P. Foster, and L. H. Ungar, "Multi-view learning of word embeddings via cca," in *Advances in Neural Information Processing Systems*, 2011, pp. 199–207.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of Workshop at ICLR*, 2013.

[15] T. H. Nguyen and R. Grishman, "Employing word representations and regularization for domain adaptation of relation extraction," in Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics(ACL), vol. 2, 2014, pp. 68–74.

[16] J. Weston, A. Bordes, O. Yakhnenko, and N. Usunier ,"Connecting language and knowledge bases with embedding models for relation extraction," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 1366–1371.