Distantly Supervised Neural Network Model for Relation Extraction

Zhen Wang, Baobao Chang, and Zhifang Sui

Key Laboratory of Computational Linguistics, Ministry of Education
School of Electronics Engineering and Computer Science, Peking University
Collaborative Innovation Center for Language Ability, Xuzhou 221009 China
wzpkuer@gmail.com, chbb@pku.edu.cn, szf@pku.edu.cn

Abstract. For the task of relation extraction, distant supervision is an efficient approach to generate labeled data by aligning knowledge base (KB) with free texts. Albeit easy to scale to thousands of different relations, this procedure suffers from introducing wrong labels because the relations in knowledge base may not be expressed by aligned sentences (mentions). In this paper, we propose a novel approach to alleviate the problem of distant supervision with representation learning in the framework of deep neural network. Our model - Distantly Supervised Neural Network (**DSNN**) - constructs the more powerful mention level representation by tensor-based transformation and further learns the entity pair level representation which aggregates and denoises the features of associated mentions. With this denoised representation, all of the relation labels can be jointly learned. Experimental results show that with minimal feature engineering, our model generally outperforms state-of-the-art methods for distantly supervised relation extraction.

1 Introduction

Relation extraction was defined as the task of generating relational facts from unstructured natural language texts. Traditional approaches to relation extraction [9, 17], using supervised learning with relation-specific examples on small hand-labeled corpora, can achieve high precision and recall. However, fully supervised paradigm is limited by the scalability of hand-labeled training data, and cannot satisfy the demand of large-scale web texts containing thousands of relations.

Distant supervision is an approach to alleviate the problem of traditional fully supervised paradigm for relation extraction. The intuition is that the training data of relations can be generated by heuristically aligning knowledge bases to free texts. Figure 1 shows the process of distant supervision to generate training examples. A *relation instance* is defined as the form $r(e_1, e_2)$, where r is the *relation name* and e_1 and e_2 are two *entity names*. An *entity mention* is a sequence of text tokens that matches the corresponding entity name in some text. A *relation mention* (*mention*) of relation instance $r(e_1, e_2)$ is a sequence of text (sentence), which contains a pair of entity mentions of e_1 and e_2. As

shown in Figure 1, the knowledge base provides two distinct relation names between an entity pair *(Barack Obama, U.S.)*. After alignment, four mentions from free texts are extracted and selected as training instances. Subsequently, previous methods often extract sophisticated lexical and syntactic features from these aligned mentions, combine them and produce extraction models which can predict new relation instances from texts.

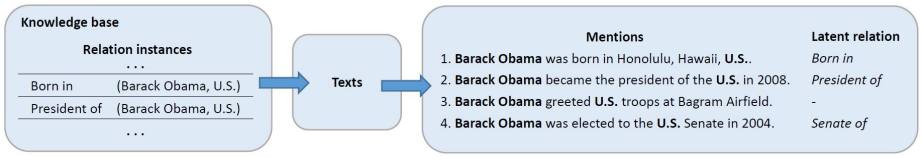

Fig. 1: Training examples generated by distant supervision with knowledge base and aligned mentions related to the entity pair *(Barack Obama, U.S.)*.

This paradigm is effective to generate large-scale training data for relation extraction. However, it suffers from three major problems.

- **Wrong labels**. Not all mentions express the relation instances from knowledge base. As shown in Figure 1, the last two sentences are not correct examples for any of the relations. Simply assuming every mention satisfies the relation instances may introduce a lot of wrong labels.
- **Multiple labels**. As shown in Figure 1, the same pair of entities may have multiple relation labels each instantiated in different scenarios, how to capture dependencies between these relations and learn them jointly is an important question.
- **Feature engineering**. Without the knowledge of what kinds of features are important for relation extraction, a huge number of lexical and syntactic features are extracted from mentions, and the performance of previous work is heavily dependent on the designing of these sophisticated features.

Concerning these challenges, in this paper, we formulate distantly supervised relation extraction from a novel perspective of representation learning, which makes the following contributions:

- We propose a Distantly Supervised Neural Network model for relation extraction. In spite of the power of neural network both in supervised and unsupervised paradigm, our model is the first neural model trained in a manner of distant supervision. The test results on the benchmark dataset show that our model outperforms previous work under the same experimental environment.
- We construct more powerful mention level representation through tensor-based transformation, which models multiple interactions in the features extracted from the mention.

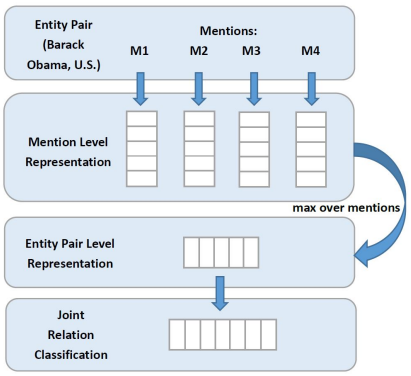

Fig. 2: The architecture of our Distantly Supervised Neural Network.

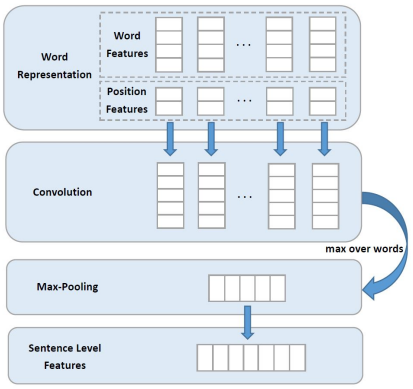

Fig. 3: The extraction framework of sentence level features.

– We propose an approach to combine the information from each mention representation and result in the entity pair level representation. This process can be regarded as a way to aggregate evident features and discard noises introducing by wrong labels.
– We apply the joint learning schema in our model to solve the multi-label problem. With the entity pair level representation, correlations of relations can be captured and the labels can be jointly learned.
– Compared with previous work that relied on a huge number of handcrafted features, our model can achieve better performance with minimal feature engineering.

The remainder of this paper is organized as follows. Section 2 introduces our model from a detailed perspective. The experimental results are presented in Section 3. Section 4 reviews the related work. Section 5 concludes this paper.

2 Distantly Supervised Neural Network

In this paper, we apply representation learning in the framework of deep neural network for distantly supervised relation extraction. In the area of NLP, it is often the case that with the help of deep neural network, representations for different levels of units, such as words, phrases, sentences and paragraphs, can be learned. With these representations, classification can be easily processed for different tasks. Inspired by the idea of representation learning, our model - Distantly Supervised Neural Network (**DSNN**) - builds the mention level representation and further entity pair level representation. With the help of these representations, all of the relation labels can be jointly learned in an efficient way.

2.1 The Neural Network Architecture

The architecture of our **DSNN** model is illustrated in Figure 2. Given a pair of entities, we extract aligned sentences from free texts. Then each sentence (mention) is represented by its feature vector, i.e. the mention level representation. In succession, we combine the information from all of the mentions to form entity pair level representation. Finally, this representation is fed into a set of binary classifiers each represents a kind of relation label. The output value of each classifier is the confidence score of the entity pair having the corresponding relation.

2.2 Mention Level Representation

The first part of our network is to transform a mention into its feature vector representation. Many approaches can be adopted in this process. For example, in the work of [15], given a sentence with parsed tree, recursive neural nets can learn the sentence vector in a bottom-up procedure. [8] proposed dynamic convolutional neural network to capture short and long-range relations when modelling sentences. [10] showed paragraph vector can be learned in a similar way as word vector. In our work, given a mention and associated entity pair, we first generate a feature vector similar to the approach proposed by [22], which combines lexical level features and sentence level features. Then we correlate these features by tensor-based transformation to form more powerful mention level representation.

Lexical Level Features Lexical level features for a mention convey the information locally embedded in the context of the given entity pair. In our work, lexical level features include the two entities, their NER tags, and the neighbor tokens of these two entities. All of these features are introduced through embeddings. After concatenation, we get the lexical level feature vector.

Sentence Level Features Sentence level features for a mention capture the global information of the whole sentence. The framework of this part is shown in Figure 3. Each word in the sentence has two kinds of features, i.e. word features and position features.

For word features, we initialize the i-th word with vector $\mathbf{c}_i$, which concatenates the embeddings for word, POS tag and NER tag. Then we adopt the window approach to generate word features:

$$\mathbf{w}_i = [\mathbf{c}_{i-d_{win}/2}^T, ..., \mathbf{c}_i^T, ..., \mathbf{c}_{i+d_{win}/2}^T]^T$$

d_{win} is the size of window, i is the current position.

For position features, we measure the distance of the current word to the two entities respectively. Then combining these two distance vector $\mathbf{d}_{i1}$ and $\mathbf{d}_{i2}$, position features are generated:

$$\mathbf{p}_i = [\mathbf{d}_{i1}^T, \mathbf{d}_{i2}^T]^T$$

Combining word features and position features, we get word representation:

$$\mathbf{x}_i = [\mathbf{w}_i^T, \mathbf{p}_i^T]^T$$

Word represented in this way only captures its local information. Inspired by [4], we adopt convolution approach to combine these local word representations of a mention into a global one. Specifically, we transform the above word representations by a linear map:

$$\mathbf{U} = \mathbf{W}_1\mathbf{X}$$

$\mathbf{X}$ forms a matrix, standing for the ensemble of the word representations in a sentence, $\mathbf{W}_1 \in \mathbb{R}^{n_1 \times n_0}$ is a linear transformation, where n_0 is the length of word representation. To find out the most useful feature in each dimension of the feature vectors in $\mathbf{U}$, a max-pooling operation is followed:

$$h_i = max\mathbf{U}(i, .) \quad 1 \leq i \leq n_1$$

Now, we get a fixed length feature vector, which captures the global information within the sentence:

$$\mathbf{h} = [h_1, h_2, ..., h_{n_1}]^T$$

Then we adopt a nonlinear transformation to learn more complex features:

$$\mathbf{s} = f(\mathbf{W}_2\mathbf{h})$$

$\mathbf{W}_2 \in \mathbb{R}^{n_2 \times n_1}$ is a linear transformation matrix, f is an activation function and we use *tanh* in our experiments. The sentence level feature vector is then defined as $\mathbf{s}$.

The final extracted feature vector $\mathbf{m}$ combines lexical level feature vector $\mathbf{l}$ and sentence level feature vector $\mathbf{s}$, formally, we adopt:

$$\mathbf{g} = [\mathbf{l}^T, \mathbf{s}^T]^T$$

$$\mathbf{m} = f(\mathbf{W}_3\mathbf{g})$$

$\mathbf{W}_3 \in \mathbb{R}^{n_4 \times n_3}$, where n_3 equals the dimension of lexical features $\mathbf{l}$ plus the dimension of sentential features $\mathbf{s}$.

Tensor-based Transformation Tensor describes relatedness between scalars, vectors and other tensors, hence it has the advantage to explicitly model multiple interactions in data. For this benefit, tensor-based methods have been used in many tasks. For example, [16] and [12] applied neural tensor layer to relate input vectors in the problems of semantic analysis and word segmentation respectively.

In our work, we adopt neural tensor layer to model the interactions of above extracted features in a mention. Figure 4 indicates this process of tensor-based transformation. As a result, the correlations among different features can be captured and more powerful mention level representation can be formed.

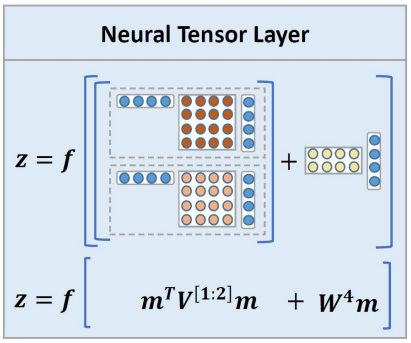

Fig. 4: The formulation of neural tensor layer. Each dashed box represents one tensor slice, which defines the bilinear form on vector $\mathbf{m}$.

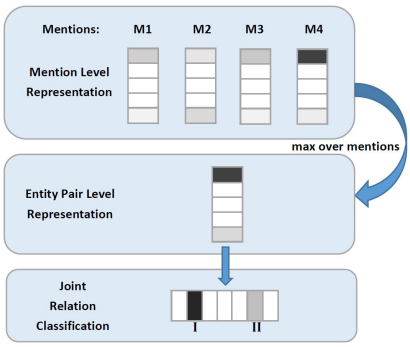

Fig. 5: The max-pooling operation to get entity pair representation. Assume the first dimension of mention representation corresponds to relation I, and the last to II. This process indicates a true relation I not relation II between the given entity pair.

Formally, a 3-way tensor $\mathbf{V}^{[1:n_5]} \in \mathbb{R}^{n_5 \times n_4 \times n_4}$ is defined to directly model the interactions. The output of a tensor product is a vector $\mathbf{a} \in \mathbb{R}^{n_5}$, whose each dimension a_i is the result of the bilinear form defined by each tensor slice $\mathbf{V}^{[i]} \in \mathbb{R}^{n_4 \times n_4}$:

$$\mathbf{a} = \mathbf{m}^T \mathbf{V}^{[1:n_5]} \mathbf{m}$$
$$a_i = \mathbf{m}^T \mathbf{V}^{[i]} \mathbf{m} = \sum_{j,k} V_{jk}^{[i]} m_j m_k$$

As indicated by equations above, in each tensor slice, the interactions are explicitly modeled by the bilinear form of the features in $\mathbf{m}$. Intuitively, we can interpret each slice of the tensor as capturing a specific type of interaction.

In our work, we augment the above tensor product with linear transformation, resulting in the final form of neural tensor layer:

$$\mathbf{z} = f(\mathbf{m}^T \mathbf{V}^{[1:n_5]} \mathbf{m} + \mathbf{W}_4 \mathbf{m})$$

where $\mathbf{W}_4 \in \mathbb{R}^{n_5 \times n_4}$ is a linear transformation matrix.

The feature vector $\mathbf{z}$ models the interactions among different features extracted from a mention, and thus can be regarded as a more powerful representation at the mention level.

In our work, we adopt a tensor factorization approach inspired by [12]. In this way, the calculation of neural tensor layer is efficient and the risk of overfitting can be alleviated.

2.3 Entity Pair Level Representation

The distinction between distantly supervised relation extraction and traditional problem of relation classification is that although from knowledge base, we can

figure out what relations the given entity pair has, we have no idea which mentions in texts truly convey these relations. Therefore, only depending on mention level representation to classify relations may introduce a lot of noises.

In our work, we combine the information embedded in all of the mentions associated with the given entity pair to form the entity pair level representation. Formally, we have the max-pooling operation along the same dimension of all the mention level representations:

$$t_i = max\mathbf{Z}(i,.) \quad 1 \leq i \leq n_5$$

$\mathbf{Z}$ is a matrix composed of all the mention representations associated with the given entity pair.

Then, we get the entity pair level representation:

$$\mathbf{t} = [t_1, t_2, ..., t_{n_5}]^T$$

We use this representation to jointly learn all of the relation labels. This process is shown in Figure 5.

The benefit of this max-pooling operation to get entity pair representation is multifold.

First, because distant supervision relation extraction is the entity pair level classification, we need to combine mention level representations all together to get global information.

Second, by max-pooling operation, each dimension of entity pair representation can hold the most important feature, which is determinant for the subsequent relation classification. Features not evident will be discarded, this can be seen as a way of feature denoising. As illustrated by Figure 5, we expect some dimension of mention level representation is crucial for a certain relation. Then, if a mention truly conveys this relation, thus has evident value in the corresponding dimension, the max-pooling operation will preserve it, and use it to indicate a true relation between the pair of entities. Otherwise, if all the mentions do not express the relation, the feature after max-pooling will not be evident either to indicate this relation.

Third, operation in this way helps us get a fixed-length feature vector, no matter how many mentions the entity pair has in texts.

At last, this operation is fairly efficient, and entity pair represented in this way is easy to calculate.

2.4 Joint Learning of Relations

Relation extraction is actually a multi-label relation classification problem. Given a pair of entities, we have no idea how many relations can be expressed. Because relation types concerned are not exclusive, it is unsuitable to regard this problem as a simple multi-classification. In our work, we treat multi-label relation classification as a set of binary relation identification problems. Concerning a certain relation type, we construct a binary classifier to determine whether

the entity pair has this relation. Formally, we adopt the logistic regression model, which has the form:

$$P(Y_i = 1|\mathbf{t}) = \frac{1}{1 + e^{-\mathbf{r}_i \cdot \mathbf{t}}}$$

where $\mathbf{t}$ is the feature vector of entity pair representation, $\mathbf{r}_i$ is the weight vector associated with the i-th relation name, $Y_i = 1$ indicates the entity pair expresses the i-th relation. This logistic function has an obvious explanation, that the higher the value of $P(Y_i = 1|\mathbf{t})$, the greater probability that the entity pair $\mathbf{t}$ indicates the i-th relation.

We can think of this procedure from a joint learning perspective. Instead of learning a specific entity pair representation for each distinct relation name, we share entity pair representation for all relations. In another word, the representation learned contains information from all of these relations, hence it can capture the correlations among different relations. Moreover, this joint learning procedure can both be efficient and avoid overfitting.

2.5 Training Criteria

Given a pair of entities x, the network with parameter θ outputs a N dimension vector $\mathbf{o}$ (N is the number of relation labels concerned in our work). The i-th component o_i corresponds to the probability that this entity pair has the i-th relation label, thus:

$$p(Y_i = 1|x, \theta) = o_i$$
$$p(Y_i = 0|x, \theta) = 1 - o_i$$

Given all our training examples:

$$T = (x^{(i)}, y^{(i)})$$

where $x^{(i)}$ denotes the i-th training entity pair, $y^{(i)}$ is the corresponding N dimension vector, and $y_k^{(i)} = 1$ indicates $x^{(i)}$ has the k-th relation.

The log likelihood with a single training sample is:

$$\sum_{k=1}^{N} logp(y_k^{(i)}|x^{(i)}; \theta)$$

And the full log likelihood of the whole training corpus is:

$$J(\theta) = \sum_{i=1}^{T} \sum_{k=1}^{N} logp(y_k^{(i)}|x^{(i)}; \theta)$$

To compute the network parameter θ, we maximize the log likelihood $J(\theta)$ using stochastic gradient ascent:

$$\theta \rightarrow \theta + \lambda \frac{\partial J(\theta)}{\partial \theta}$$

where λ is the learning rate controlling the step of gradient ascent.

Remark	Choice
Window size	$d_{win} = 3$
Word embedding dimension	$n_{word} = 50$
POS tag dimension	$n_{pos} = 20$
NER tag dimension	$n_{ner} = 10$
Distance dimension	$n_{dis} = 20$
Hidden layer 1	$n_1 = 200$
Hidden layer 2	$n_2 = 100$
Hidden layer 3	$n_4 = 200$
Neural tensor layer	$n_5 = 100$
Learning rate	$\alpha = 10^{-3}$

Table 1: Hyper parameters of our model DSNN.

Models	P	R	F1
Mintz et al., 2009	26.17	22.67	24.29
Hoffmann et al., 2011	32.05	24.05	27.48
Surdeanu et al., 2012	**32.92**	20.56	25.32
DSNN	29.81	**33.45**	**31.53**

Table 2: Results at the highest F1 point in the precision-recall curve.

3 Experiments

To evaluate the performance of our proposed approach, we conduct three experiments. The first one compares our **DSNN** with previous landmark methods on the public dataset. The second is carried out in our **DSNN** framework with or without the neural tensor layer, the experiment is designed to show the ability of tensor operation to get more powerful mention level representation. The last experiment displays the distribution of entity pair level representations after training, the result shows the effectiveness of this representation to capture relation information.

3.1 Dataset

We evaluate our algorithm on the widely-adopted dataset developed by [13]. In this dataset, Freebase was used as the distant supervision source and New York Times (NYT) was selected as the text corpus. The Freebase relation types concerned in this dataset focus on four categories of entities: "people", "business", "person" and "location", and NYT data contains over 1.8 million articles between January 1, 1987 and June 19, 2007, which are partitioned into training set and testing set. After alignment, we get 51 kinds of relation labels.

3.2 Experimental Results

The hyper parameter setting of our **DSNN** are reported in Table 1. The POS tags and NER tags are automatically generated by CoreNLP[1]. We initialize the word embeddings with pre-trained distributed vectors devoted by [4]. The embeddings of POS tags, NER tags and other weights in our model are randomly initialized.

[1] http://nlp.stanford.edu/software/corenlp.shtml

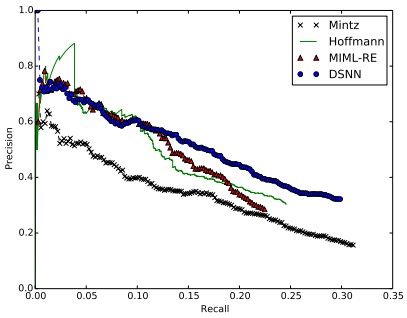

Fig. 6: Results comparison on the Riedel dataset.

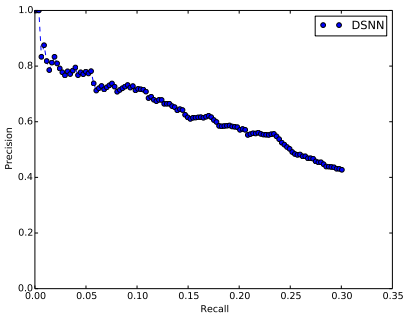

Fig. 7: Results on instances with more than 1 mention.

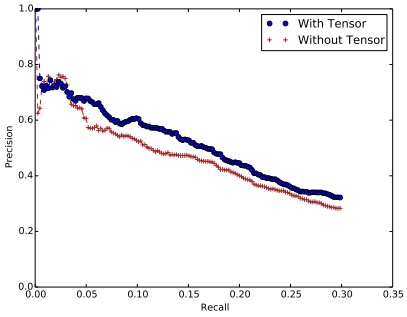

Fig. 8: The comparison of results with or without the neural tensor layer.

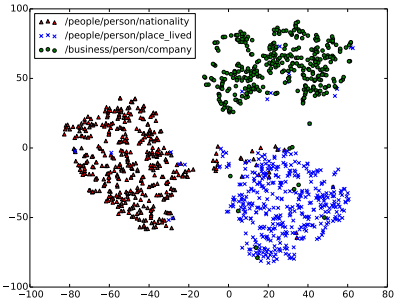

Fig. 9: The feature vectors of entity pair representation are visualized using t-SNE.

The first part of our experiments is to compare our work with previous landmark methods [11, 7, 18]. The results of previous work can be reproduced by the open source code[2] developed by [18].

Figure 6 indicates that our **DSNN** model generally outperforms the current state-of-the-art results on the Riedel dataset. As shown in the figure, in the area of low recall, our model performs competitively with previous methods, and as recall gets larger, the performance of **DSNN** is significantly better than all of the competitors. Therefore, our approach gets largest precision-recall area and highest F1 score, which is shown in Table 2.

As Figure 6 indicates, our approach can maintain a fairly high precision even when recall is larger. Therefore, when it is the situation that we need to pick out a large number of candidate relation instances with relatively high precision, our **DSNN** model is the most suitable choice.

[2] http://nlp.stanford.edu/software/mimlre.shtml

We can explain this performance improvement via the distinction between our model and previous ones. Previous approaches more relied on the sophisticated features extracted from mentions, and resulted in a large number of sparse features. As a consequence, if testing instances can reproduce the features in the training procedure, the probability of right prediction will be fairly high. However, because of the sparsity, when the features do not match between testing and training, the generalization ability of feature-based methods will be badly hurt. With the framework of neural network, our model discards the sophisticated features, learns the entity pair distributed representation. This can be regarded as the process of letting data decide what are important features for our problem. Because of this advantage, feature sparsity is alleviated and the generalization ability is enhanced.

To further clarify the ability of our entity pair representation to aggregate information from mentions, we conduct experiment on relation instances with more than 1 mention aligned in texts, with the result shown in Figure 7. The precision-recall performance in this subset is highly improved compared to the initial dataset. Therefore, more mentions give more confidence on the prediction of relations with our approach. With this knowledge, if we are more concerned with the precision of relation instances extracted, we can resort to the number of mentions for further improvement.

The second part of our experiments is carried out in our **DSNN** framework with or without the neural tensor layer. The comparison of results is illustrated in Figure 8. As shown in this figure, mention level representation learned by tensor-based transformation is more powerful and has a good influence on the performance of distant supervision relation extraction.

The third part of our experiments is to demonstrate the effectiveness of our entity pair representation. Using t-SNE[3], the feature vectors of entity pair representation can be well painted as indicated in Figure 9. In this experiment, we take three kinds of relations into consideration, and each relation contains a large number of entity pairs holding this relation. As a result of Figure 9, entity pair instances with the same relation will be gathered together, which shows the ability of our entity pair representation to convey relation information.

4 Related Work

4.1 Distant Supervision

Distant supervision is first introduced by [5], who focused on the field of bioinformatics. Since then, this approach scaled to many other fields [14, 2, 20]. As for relation extraction, [11] adopted Freebase to distantly supervise Wikipedia corpus. This work was dependent on the basic assumption that if an entity pair participates in a relation in the knowledge base, all sentences from texts that matched the facts are labeled by that relation name. As shown in Figure 1, this procedure may introduce a lot of wrong labels. To avoid this problem, [13]

[3] http://lvdmaaten.github.io/tsne/

relaxed the distant supervision assumption with multi-instance learning framework, which replaced all sentences with at least one sentence expressing the relation. Then [7] proposed multi-label circumstance to enrich previous work. [18] advanced with a novel approach to jointly model all the mentions in texts and all the relation labels in knowledge base, resulting in a multi-instance multi-label learning framework for relation extraction. [19] proposed a generative approach to model the heuristic labeling process in order to reduce wrong labels. [6] applied matrix completion with convex optimization to tackle the sparsity and noise challenges of distant supervision. [1] provided partial supervision using a small number of carefully selected examples. [21] resolved the noise features and exploited sparse representation to solve the problem.

4.2 Representation Learning

Representation learning is a paradigm to capture the underlying explanatory factors hidden in the observed data and make learning less dependent on feature engineering. In the area of NLP, with a powerful representation, classification can be easily processed. Therefore, the idea of representation learning has been scaled to many tasks [3, 4, 15].

[22] made use of lexical level features and sentence level features to form mention level representation, then with a softmax classifier the problem of relation classification was well solved. We get inspiration from their work to form the features extracted from mentions. However, relation classification is a totally supervised problem, which has no trouble of wrong labels and multi-label circumstance. With the entity pair representation and joint learning schema, our model can solve these challenges pretty well. Moreover, we enrich their mention features with tensor-based transformation, resulting in a more powerful mention level representation.

5 Conclusion

In this paper, we showed that distant supervision for relation extraction can be formulated with the framework of deep neural network. In our model **DSNN**, interactions of extracted features in a mention are captured to construct more powerful mention level representation. Then the pair of entities combines the information from all the aligned mentions, and forms the representation at the level of entity pair. This process can aggregate evident features from different mentions, as well as discard noises, which are introduced by the paradigm of distant supervision. The learned representation then is used to jointly learn all of the relation labels. Moreover, with the framework of deep neural network, the heavy job of feature engineering is much alleviated. Experiments show the effectiveness of the mention level tensor-based transformation and the ability of the entity pair level representation to capture relation information. Moreover, we compare our model with state-of-the-art results on the benchmark dataset, and demonstrate our approach achieves improvements on performance.

Acknowledgments

This research is supported by National Key Basic Research Program of China (No.2014CB340504) and National Natural Science Foundation of China (No.613-75074,61273318). The contact authors of this paper are Baobao Chang and Zhifang Sui.

References

1. Angeli, G., Tibshirani, J., Wu, J.Y., Manning, C.D.: Combining distant and partial supervision for relation extraction. In: Proc. The 2014 Conference on Empirical Methods on Natural Language Processing (2014)
2. Bellare, K., McCallum, A.: Learning extractors from unlabeled text using relevant databases. In: Sixth international workshop on information integration on the web (2007)
3. Bengio, Y., Ducharme, R., Vincent, P., Janvin, C.: A neural probabilistic language model. The Journal of Machine Learning Research 3, 1137–1155 (2003)
4. Collobert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th international conference on Machine learning. pp. 160–167. ACM (2008)
5. Craven, M., Kumlien, J.: Constructing biological knowledge bases by extracting information from text sources. In: ISMB. vol. 1999, pp. 77–86 (1999)
6. Fan, M., Zhao, D., Zhou, Q., Liu, Z., Zheng, T.F., Chang, E.Y.: Distant supervision for relation extraction with matrix completion. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. vol. 1, pp. 839–849 (2014)
7. Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., Weld, D.S.: Knowledge-based weak supervision for information extraction of overlapping relations. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. pp. 541–550. Association for Computational Linguistics (2011)
8. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (2014)
9. Kambhatla, N.: Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In: Proceedings of the ACL 2004 on Interactive poster and demonstration sessions. p. 22. Association for Computational Linguistics (2004)
10. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. arXiv preprint arXiv:1405.4053 (2014)
11. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: 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. pp. 1003–1011. Association for Computational Linguistics (2009)
12. Pei, W., Ge, T., Baobao, C.: Maxmargin tensor neural network for chinese word segmentation. In: Proceedings of ACL (2014)

13. Riedel, S., Yao, L., McCallum, A.: Modeling relations and their mentions without labeled text. In: Machine Learning and Knowledge Discovery in Databases, pp. 148–163. Springer (2010)
14. Snow, R., Jurafsky, D., Ng, A.Y.: Learning syntactic patterns for automatic hypernym discovery. Advances in Neural Information Processing Systems 17 (2004)
15. Socher, R., Huval, B., Manning, C.D., Ng, A.Y.: Semantic Compositionality Through Recursive Matrix-Vector Spaces. In: Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2012)
16. Socher, R., Perelygin, A., Wu, J.Y., Chuang, J., Manning, C.D., Ng, A.Y., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the conference on empirical methods in natural language processing (EMNLP). vol. 1631, p. 1642. Citeseer (2013)
17. Suchanek, F.M., Ifrim, G., Weikum, G.: Combining linguistic and statistical analysis to extract relations from web documents. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 712–717. ACM (2006)
18. Surdeanu, M., Tibshirani, J., Nallapati, R., Manning, C.D.: Multi-instance multi-label learning for relation extraction. In: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. pp. 455–465. Association for Computational Linguistics (2012)
19. Takamatsu, S., Sato, I., Nakagawa, H.: Reducing wrong labels in distant supervision for relation extraction. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. pp. 721–729. Association for Computational Linguistics (2012)
20. Wu, F., Weld, D.S.: Autonomously semantifying wikipedia. In: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management. pp. 41–50. ACM (2007)
21. Zeng, D., Lai, S., Wang, X., Liu, K., Zhao, J., Lv, X.: Distant supervision for relation extraction via sparse representation. In: Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, pp. 151–162. Springer (2014)
22. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J.: Relation classification via convolutional deep neural network. In: Proceedings of COLING. pp. 2335–2344 (2014)