Relation Classification in Scientific Papers Based on Convolutional Neural Network*

YIN Zhongbo[1], WU Shuai[1], YIN Yi[2], LUO Wei[1]✉**, LUO Zhunchen[1], TAN Yushani[1], JIAO Xiangyu[3], and WANG Dong[4]

[1] Academy of Military Science , Beijing 100142, China
[2] Haiyan County Xiangyang Primary School, Jiaxing 314300, China
[3] JiLin University, Changchun 130012, China
[4] Information Engineering University, Zhengzhou 450000, China
lwowen79@gmail.com

Abstract. Scientific papers are important for scholars to track trends in specific research areas. With the increase in the number of scientific papers, it is difficult for scholars to read all the papers to extract emerging or noteworthy knowledge. Paper modeling can help scholars master the key information in scientific papers, and relation classification (RC) between entity pairs is a major approach to paper modeling. To the best of our knowledge, most of the state-of-the-art RC methods are using entire sentence's context information as input. However, long sentences have too much noise information, which is useless for classification. In this paper, a flexible context is selected as the input information for convolution neural network (CNN), which greatly reduces the noise. Moreover, we find that entity type is another important feature for RC. Based on these findings, we construct a typical CNN architecture to learn features from raw texts automatically, and use a softmax function to classify the entity pairs. Our experiment on SemEval-2018 task 7 dataset yields a macro-F1 value of 83.91%, ranking first among all participants.

Keywords: Relation Classification, Convolution Neural Network, Scientific Paper Modeling, Entity Type, Context Scope

1 Introduction

As the main source of recording new technology, mining scientific papers is a common method to track the progress of research. In most cases, scholars cannot read all of papers to extract noteworthy aspects in their research field. Information extraction (IE) , including named entity recognition (NER) and relation extraction (RE), is a primary NLP method for analyzing scientific papers. This paper focuses on relation classification (RC, sub-method of RE) between entities in scientific papers. Relation classification can be used to predict the relations

* First three authors have equal contribution to this paper.
** Corresponding Author.

between entity pairs [16]. For example, the following sentence contains an example of the **Part-Whole** relation between entity pairs : **corpus-sentences**.

The <e1>corpus</e1> consists of seven hundred semantically neural <e2> sentences</e2>.

Most traditional RC methods rely on handcrafted features or additional NLP tools to extract lexical features [15, 6, 12]. However, this is time consuming and leads to error propagation. In addition, the traditional RC approaches use a large number of pairwise similarity features (eg, matching characters, word n-grams or co-occurrence subsequences) to measure textual semantic similarity. But these features may be difficult to represent syntactic information, which is more important for analyzing relations between entities.

In order to represent the syntactic information between entities, some DNN-based methods have been proposed and achieved remarkable results [14, 3, 8]. Since SemEval-2010 task 8 provides a benchmark for classifying relations between target nominals in a given sentences set, quite a number of DNN-based RC methods have been optimized and developed. Daojian's work [16] is the most representative progress, it builds a new CNN architecture for RC. They use convolution networks to extract lexical and sentence level features, then feed them into softmax classifier. Furthermore, Qin's work [8] proposes a stacked neural network model in which CNN is used for sentence modeling, and a feature extraction collaborative gated neural network (CGNN) is proposed.

This paper is based on the work of Pengda [9] which deals RC with a CNN architecture to automatically control feature learning from raw sentences and minimize the application of external toolkits and resources. As Daojian [16] proposes an entity position feature to highlight the entity's function in RC, it uses several additional dimensions following each word's embedding to represent the relative distances from the current word to the first and second entity. Since the position feature's dimension (e.g. 5 or 10) is much smaller than word embedding's (e.g. 100 or 300), it will disappear during the excessive training times.

Thus, we use Pengda's Entity Tag Feature [9] to emphasize the entity pair information, which uses the tagged words (*<e1s>, <e1e>, <e2s>, <e2e>*) to indicate start and end position features of entities. Unlike Daojian's work, these tag words are represented as independent word embedding to avoid the disappeared problem.

To the best of our knowledge, most pervious DNN-based methods used entire sentence's information as the context for extracting features for RC [13, **?**]. However, our goal is RC rather than sentence classification. Even if using position or entity tag feature to highlight the entity function, it still exists a lot of noise for long sentences. In pervious works, many researchers just used the context between two entities, which got a remarkable performance promotion [9]. However, sometimes, when there are few context words between two entities, semantic information cannot be detected. Therefore, we propose a flexible context scope selection algorithm that has achieved significant improvements in experiment.

What's more, inspired by Schwartz's work [10] , we consider the entity type to be a useful feature for RC. For example, the **Compare** relation may mean that ***X is compared to Y***, where X or Y have the same entity type, such as two technologies or two models. Therefore, we attempt to include the type feature into feature set and achieve better RC performance.

2 Methodology

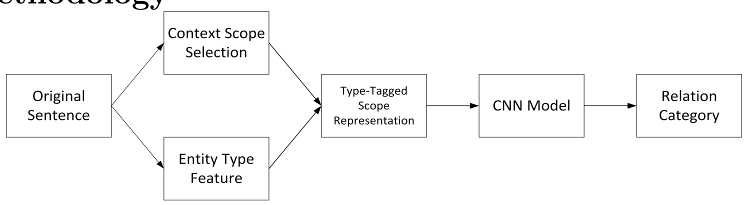

Fig. 1. Relation Classification Framework

Our RC framework is shown in Figure 1. First, we select the scope of context words and construct an entity type dictionary to map the entity into a specific type item. To represent the tagged type and scope information, we convert all of them into word embeddings. Then, all the embeddings will be transmitted to three convolution networks with kernel sizes are 3, 4 and 5 respectively. Finally, the three convolution outputs are pooled into the same dimensional vector, which will be concatenated as an input to the softmax relation classifier.

2.1 Context Scope Selection

Most of the existing DNN-based methods use entire sentence context words embedding as input. As the following sentence, there is a **Compare** relation between ***bag-of-words method*** and ***segment order-sensitive methods***. However, the sub sentences before *<e1s>* and after *<e2e>* have little relevance to the target relation category. Notwithstanding, the sub sentence between *<e1s>* and *<e2e>* has more relevance information to the target relation.

Further, in their optimum configuration, <e1s>bag-of-words method<e1e> are equivalent to <e2s> segment order-sensitive methods <e2e> in terms of retrieval accuracy, but much faster.

As a result, we propose a flexible and suitable context scope selection algorithm. Firstly, we select the words of entity pair's names and the intermediate words as the major context. Then, we notice that the major context feature is too weak when there are few intermediate words between entity pair. Actually, we got the worst classification performance while the intermediate words number smaller than 2 in experiment. In such a case, we extend context scope before the first entity and after the second entity while the number of words in the middle is smaller than 2. The specific **Context Selection** section's procedure in Figure 1 is depicted in Figure 2. We use the Stanford NLP group's parser[5] to parse the sentence with compositional vector Grammar and select the minimal sub tree (sentence) containing the major context as the extend context [5].

[5] https://nlp.stanford.edu/software/lex-parser.html

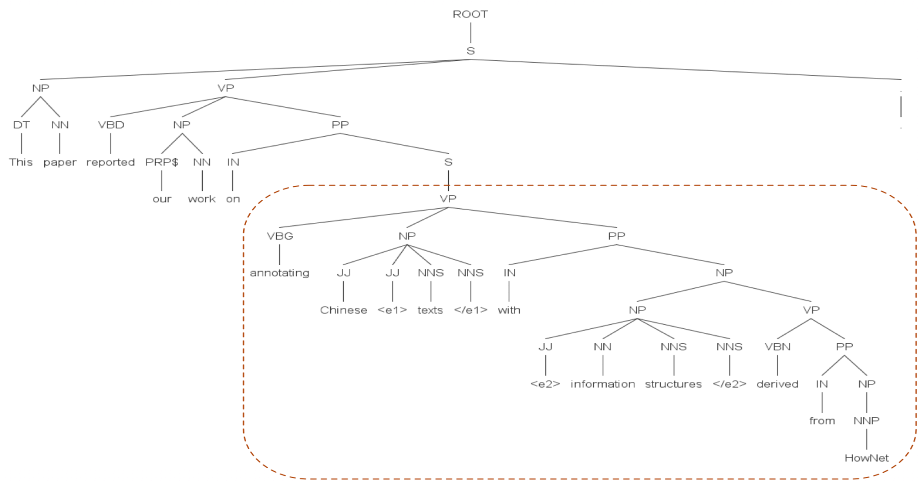

Fig. 2. Context Selection Procedure

2.2 Entity Type Feature

Some relation categories are closely related to entity types. For instance, the **Compare** relation is usually used to describe the similarity between two entities which have the same type, such as two ***models*** or two ***technologies***. Moreover, at least one entity of the **Result** relation is ***Measurement*** type such as *accuracy*, *precision* and *recall*. Therefore, the RC performance will be improved if we include the entity type into feature set.

To determine the entity type, we build an entity type vocabulary in computational linguistic domain. This vocabulary is based on the ACL Anthology Reference Corpus [2] which contains lots of scholarly publications about computational linguistics. Furthermore, Behrang [7] manually annotated thousands of term's type in this ACL Corpus. In the Behrang's work, terms are annotated with one of the 7 types: **technology, system, language resources, language resources (specific product), model, measurement and other**. We use these term-type pairs to build the original entity type vocabulary. Then, the original vocabulary is filtered and expanded by some prior language knowledge.

Table 1. Samples of Entity Type Vocabulary

Entity	Type	Entity	Type
human-computer interaction	technology	WordNet	language resources product
bayesian network	technology	Reuters-21578	language resources product
NLTK	tool	n-gram model	model
Stanford Core NLP	tool	maximum entropy model	model
dictionary	language resources	BLEU	measurement
syntactic rule	language resources	F-score	measurement

The original vocabulary contains 2583 manual annotated entity-type pairs. It contains 762 overlap pairs and 770 **Other** pairs. After filtering out these pairs, we receive a vocabulary containing 1024 entity-type pairs. Then we use

SemEval-2018 task 7.1.1 training set [11] to expand the vocabulary. We have two rules. First, we believe that the entity pair for **Compare** relation have the same entity type. Second, there are at least one entity's type is *Measurement* in **Result** relation's entity pair. After expanding by these rules, we receive a vocabulary containing 1654 entity-type pairs. Some instances in the vocabulary are listed in Table 1.

2.3 CNN Architecture

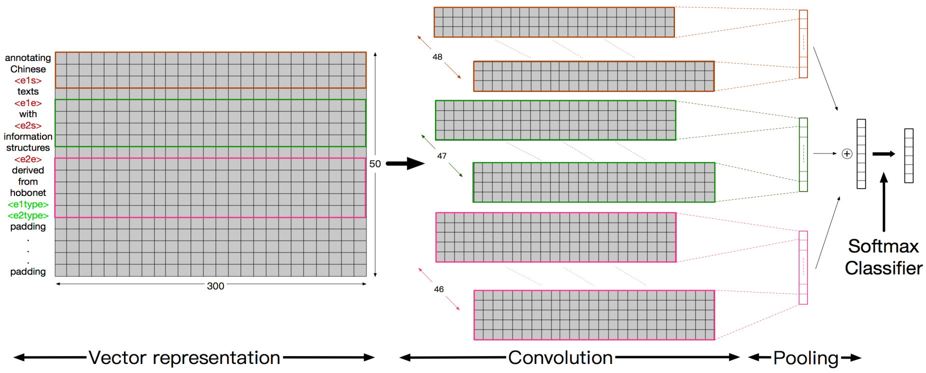

Fig. 3. Convolutional Neural Network for Relation Classification

As depicted in Figure 3, our relation classification convolutional neural network (RCNN) structure contains four steps: vector representation, convolution, pooling and classifying. In the vector representation step, all of the words selected in context scope selection and entity type setting parts are mapped into 300-dimensions vector. At the same time, we keep only the last 50 words of the sentence if the sentence contains more than 50 words. We add additional padding vector while the sentence's words smaller than 50. In the convolution step, the selected vector are delivered into three CNNs whose kernel size are 3, 4 and 5 respectively. It means that all the 3-grams, 4-grams and 5-grams features will be considered. To normalize different length of each input sentence, the convolution output will be pooled into the same dimensional vector space. Finally, we use the multi-class classifier softmax to classify the instance into a specific relation category.

Table 2. Hyper-parameters Used in Our Experiment

Embedding-dim	Batch-size	Dropout	Learning-rate	Classifier	Max-sentence	Kernel-size
300	30	0.5	1e-4	Softmax	50	3,4,5

The experiment settings are listed in Table 2, we use the Wikipedia general English 300-dimensional embedding which contains 408 million words[6] to represent the words in the context. The experiments results show that the hyper-parameters in the Table 2 gains the most good performance.

[6] https://www.cs.york.ac.uk/nlp/extvec/wiki_extvec.gz

3 Experiment

3.1 Dataset

We use SemEval-2018 task7.1.1 dataset [11] which contains titles and abstracts of scientific papers in computational linguistic domain. Six predefined semantic relations are manually annotated in this dataset. It contains 1248 training and 355 testing examples. As in Table 3, in the training set, every instance is classified into one of the following relations: Usage, Result, Model-Feature, Part-Whole, Topic, Compare.

Table 3. Samples of Semantic Relations

Relation	Explanation
Part-Whole	X is a part or component of Y
Model-Feature	X is a model, feature or characteristic of Y
Result	X yields Y (e.g. improvement or decrease)
Usage	X (e.g. method, tool) is used for Y (e.g. task, data, system)
Compare	X is compared to Y (e.g. two systems, feature sets or results)
Topic	X (e.g. author, paper) puts forward Y (e.g. san idea, an approach)

3.2 Effect of New Context Scope

Table 4. Results for Changing Scope and Adding Type Feature

Feature	Precision	Recall	macro-F1
entire sentence scope	64.56	57.24	60.68
+type feature	65.78	59.24	62.34
major context scope	76.50	63.58	69.44
+type feature	77.06	66.32	71.29
flexible context scope	87.10	77.39	81.96
+type feature	**89.01**	**78.23**	**83.27**

The results in Table 4 are acquired in different context scopes and whether using type feature. The precision is higher than recall in each experiment. Additionally, major context scope's classification performance is better than entire sentence scope. It shows that the major context (entity pair names and words between them) contains more accurate and cleaner semantic information than entire sentence context as for RC. Further more, our flexible context achieve a better performance than major context which scores the best macro-F1: 83.27. By analyzing the specific instances of these experiments, we find that many wrong predicted long sentence instances in entire sentence context (context1) have been corrected in major context (context2) and our flexible context (context3) experiment as in Table 5. Additionally, some wrong predicted instances in context2 which are few words between entity pairs have been corrected in context3.

Table 5. Context Scope Selection Experiment Result Examples

Sentence	Scope	Prediction	True/False
We present an implementation of the model based on finite-state models, demonstrate the <e1s> model's <e1e> ability to significantly reduce <e2s> character and word error rate <e2e>, and provide evaluation results involving automatic extraction of translation.	context1	Compare	False
	context2	Result	True
	context3	Result	True
With its higher-order <e1> representations </e1> of <e2> contexts </e2> , TDL analyzes and describes the inher- ently inter-sentential nature of quantification and anaphora in a strictly lexicalized and compositional manner .	context1	Part-Whole	False
	context2	Part-Whole	False
	context3	Model-Feature	True

3.3 Effect of Entity Type Information

As in Table 4, all of the comparison experiments get promotion when we add entity type embedding to the original feature set. As in Table 6, the wrong prediction is corrected when the entity type feature has been included into feature set. The experiment performance promotion prove that the type feature we proposed is helpful for identifying the semantic relation between two entities in RC.

Table 6. Type Feature Experiment Result examples

Sentence	Type	Prediction	True/False
Experiments with the TREC2003 and TREC- 2004 QA tracks indicate that <e1s> rankings <e1e> produced by our metric correlate highly with <e2s>official rankings<e2e>, and that POURPRE outperforms direct application of existing metrics.	None	Model-Feature	False
	Yes	Compare	True

As depicted in Figure 4, every word in the selected input sentence is mapped into embedding before deliver to CNN.As in word representation step, except the general word embedding, the entity tag (*<e1s>, <e1e>, <e2s>, <e2e>*) also mapped into embeddings. Additionally, the last two green embeddings are used to encode entity type features. For a bit more promotion, we use entity tag feature to substitute entity position feature (*<e1s>, <e1e>, <e2s>, <e2e>*) to explore the representation improvements. RCNN-ST means relation classification

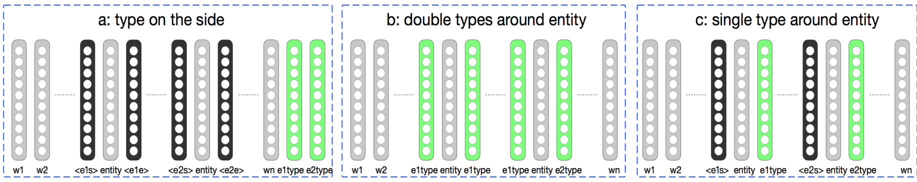

Fig. 4. Type Feature Position Setting

CNN in section 3.1 with flexible context scope and type feature. As Figure 4, we develop three schemes to distribute the type feature on the sentence side(RCNN-STa), double around entity(RCNN-STb) and single around entity(RCNN-STc). For these adjustments, we get the classification performance as Table 7. To our suprise, we get much promotion while the type position adjust to both side and single side around entity.

Table 7. Results for Testing Type Position Feature

RCNN-STa	RCNN-STb	RCNN-STc
83.27	83.51	83.91

3.4 Comparison Experiment

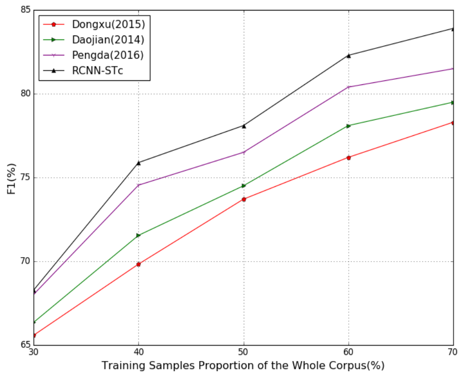

Fig. 5. Results for Compare with Other Methods

This experiment compares our RCNN-STc approach with three state-of-art neural models (Dongxu(2015)[17], Daojian(2014)[16], Pengda(2016)[9]). The feature set they used is listed in Table 8. The results are shown in Figure 5. As we can see, our RCNN-STc approach gets the best results in every training samples proportion.

3.5 Generalization on Different Datasets

We use two more different datasets to test our method. One is the dataset provided by SemEval-2010 Task 8. There are 9 directional relations and an additional **other** relation, resulting in 19 relation classes in total[4]. The second dataset is a revision of MIML-RE annotation dataset, provided by Gabor[1]. They use both the 2010 and 2013 KBP official document collections, as well as a July 2013 dump of Wikipedia as the text corpus for annotation. We test our

Table 8. Feature Sets of Comparison Method

Classifier	Feature sets
Daojian(2014)	word embedding(dim=50), position feature, Wordnet
Dongxu(2015)	word embedding(dim=300), position feature, position indicators
Pengda(2016)	word embedding(dim=300), entity tag
RCNN-STc	word embedding(dim=300), entity tag, context selection, entity type

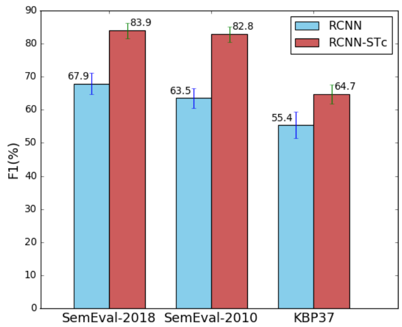

Fig. 6. Results for Comparison in Different Datasets

RCNN and RCNN-STc on these datasets. The relation classification results is depicted in Figure 6.

4 Conclusion

The contributions of this paper can be summarized as follows: Firstly, we construct a typical CNN architecture for RC without sophisticated NLP preprocessing. Secondly, we explore a flexible scope context input for CNN, which extremely reduce the useless context's noise influence. Thirdly, we build an entity type vocabulary and add the type embedding into feature set, which enhance the entity's semantic representation consequently. At last, we discuss the way to feed CNN with type feature position embedding, which transmitting more original sequence information. Finally, our proposed method gets 83.91% macro-F1 value and ranks first in SemEval-2018 task 7.

Acknowledgements

Firstly, we would like to thank Bin Mao, Changhai Tian and Yuming Ye for their valuable suggestions on the initial version of this paper, which have helped a lot to improve the paper. Secondly, we want to express gratitudes to the anonymous reviewers for their hard work and kind comments, which will further improve our work in the future. Additionally, this work was supported by the National Natural Science Foundation of China (No. 61602490)

Reference

1. Angeli, G., Tibshirani, J., Wu, J., Manning, C.D.: Combining distant and partial supervision for relation extraction. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1556–1567 (2014)

2. Bird, S., Dale, R., Dorr, B.J., Gibson, B., Joseph, M.T., Kan, M.Y., Lee, D., Powley, B., Radev, D.R., Tan, Y.F.: The acl anthology reference corpus: A reference dataset for bibliographic research in computational linguistics (2008)
3. Guo, J., Che, W., Wang, H., Liu, T., Xu, J.: A unified architecture for semantic role labeling and relation classification. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. pp. 1264–1274 (2016)
4. Hendrickx, I., Kim, S.N., Kozareva, Z., Nakov, P., Ó Séaghdha, D., Padó, S., Pennacchiotti, M., Romano, L., Szpakowicz, S.: Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In: Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions. pp. 94–99. Association for Computational Linguistics (2009)
5. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of the 41st annual meeting of the association for computational linguistics (2003)
6. Kozareva, Z.: Cause-effect relation learning. In: Workshop Proceedings of TextGraphs-7 on Graph-based Methods for Natural Language Processing. pp. 39–43. Association for Computational Linguistics (2012)
7. QasemiZadeh, B., Schumann, A.K.: The acl rd-tec 2.0: A language resource for evaluating term extraction and entity recognition methods. In: LREC (2016)
8. Qin, L., Zhang, Z., Zhao, H.: A stacking gated neural architecture for implicit discourse relation classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 2263–2270 (2016)
9. Qin, P., Xu, W., Guo, J.: An empirical convolutional neural network approach for semantic relation classification. Neurocomputing **190**, 1–9 (2016)
10. Schwartz, R., Reichart, R., Rappoport, A.: Minimally supervised classification to semantic categories using automatically acquired symmetric patterns. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. pp. 1612–1623 (2014)
11. SemEval2018: Semeval2018, https://competitions.codalab.org/competitions/17422
12. 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)
13. Xu, K., Feng, Y., Huang, S., Zhao, D.: Semantic relation classification via convolutional neural networks with simple negative sampling. arXiv preprint arXiv:1506.07650 (2015)
14. Yin, Z., Luo, Z., Luo, W., Bin, M., Tian, C., Ye, Y., Wu, S.: IRCMS at SemEval-2018 task 7 : Evaluating a basic CNN method and traditional pipeline method for relation classification. In: Proceedings of The 12th International Workshop on Semantic Evaluation. pp. 811–815. Association for Computational Linguistics, New Orleans, Louisiana (Jun 2018). https://doi.org/10.18653/v1/S18-1129, https://www.aclweb.org/anthology/S18-1129
15. Yin, Z., Tang, J., Ru, C., Wei, L., Luo, Z., Ma, X.: A semantic representation enhancement method for chinese news headline classification (2017)
16. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J.: Relation classification via convolutional deep neural network. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. pp. 2335–2344 (2014)
17. Zhang, D., Wang, D.: Relation classification via recurrent neural network. arXiv preprint arXiv:1508.01006 (2015)