Chinese Geographical Knowledge Entity Relation Extraction via Deep Neural Networks

Shengwu Xiong, Jingjing Mao, Pengfei Duan* and Shaohao Miao
Computer Science and Technology, Wuhan University of Technology, 122 Ruoshi Road, Wuhan, Hubei, China

**corresponding author*

Keywords: DNNs, Relation Extraction, Chinese Geographical Knowledge.

Abstract: Aiming at the problem of complex relation pattern and low relation extraction precision in the unstructured free text, in this paper, a novel extraction model for Chinese geographical knowledge relation extraction using a real end-to-end deep neural networks (DNNs) is proposed. The proposed method is a fusion DNNs consisting of one convolutional neural networks and two neural networks, which contains word feature, sentence feature and class feature. For the experiments, we construct geographic entity relation type system and corpus. We achieve a good performance with the averaged overall precision of 96.54%, averaged recall of 92.99%, and averaged F value of 94.56%. Experimental results confirm the superiority of the proposed Chinese geographical knowledge relation extraction method. The data of this paper can be obtained from http://nlp.webmeteor.cn.

1 INTRODUCTION

Relation extraction is an important but complex task, which is normally used as an essential step in a wide range of natural language processing applications such as construction of question-answering and knowledge corpus. However, Chinese relation extraction often relies on a set of complex natural language processing tasks, including Chinese word segmentation, named entity recognition, part of speech (POS) tagging, etc. Hence Chinese relation extraction is a more challenging problem.

The complete and high quality knowledge corpus determines the level of human like intelligence. However, there is a lack of geographic knowledge under construction and open geographic corpus, as well as few studies on Chinese geographical knowledge entity relation extraction.

To solve the problem concerning Chinese relation extraction, a number of deep learning approaches including multilayer neural networks, recurrent neural networks, long-short term memory, etc., have been recently applied. Machine learning has been successfully applied in many different natural language processing tasks such as part of speech tagging (Santos et al.,2014), sentiment analysis (Kim,2014; Santos et al.,2014), sentence classification (Kalchbrenner et al.,2014), semantic role labelling (Collobert et al.,2014), semantic matching (Hu et al.,2015), etc. These methods based on deep neural networks are in need of some complex features, such as named entity, dependency trees, and etc. However, the error in extracting complex features may accumulate, and features designed artificially are not certainly suitable for Chinese geographical knowledge entity relation extraction, and new tasks needs to design new features artificially.

To address the aforementioned problem, this paper proposes an extraction technique for Chinese geographical knowledge relation extraction, using a real end-to-end Deep Neural Networks coupled with word features, sentence features and class features after constructing geographic entity relation type system and corpus. Firstly, we use a real-valued vector to represent a character. Then we use a neural network to extract the local features, utilize a convolutional neural network to extract global features, adopt a neural network to extract the class features at the same time. Finally, we assemble the three features head-to-tail in sequence, as the input of fully connected layer, to predict the label of character.

The main contributions of this paper are as follows: (1) We construct geographic entity relation type system and corpus artificially, and we make our data public for others. (2) The relation extraction doesn’t need complicated NLP pre-processing. (3) We establish a real end-to-end deep neural network coupled with word features, sentence features and class features.

Xiong S., Mao J., Duan P. and Miao S.
Chinese Geographical Knowledge Entity Relation Extraction via Deep Neural Networks.
DOI: 10.5220/0006109700240033
In *Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017)*, pages 24-33
ISBN: 978-989-758-220-2

The remainder of this paper is organized as follows. Section 2 reviews related work and Section 3 gives a detailed account of the proposed deep neural network and the features which have been used in this paper. Section 4 details the geographical corpus we constructed. The experimental results are introduced in Section 5. We finally conclude our work in Section 6.

2 RELATED WORK

Recently, deep learning has become an attractive area for multiple applications. It has made major breakthroughs in the field of computer vision and speech recognition, and natural language understanding is another area in which deep learning is poised to make a large impact over the next few years (LeCun et al.,2015). Among the different deep learning strategies, convolutional neural networks have been successfully applied to different NLP task such as POS tagging, sentiment analysis, semantic role labelling, etc.

Some people have applied deep learning on relation extraction. Socher et al. (2012) tackle relation classification using a recursive neural network that assigns a matrix-vector representation to every node in a parse tree. The representation for longer phrases is computed bottom-up by recursively combining the word according to the syntactic structure of the parse tree. Mikolov et al. (2013) found that the vector-space word representations are surprisingly good at capturing syntactic and semantic regularities in language, and that each relationship is characterized by a relation-specific vector offset. For example, the male/female relationship is automatically learned, and with the induced vector representations, “King – Man + Woman” results in a vector very close to “Queen”. It is an effective theoretical support for instance relation and parallel relation extraction. Zeng et al. (2014) proposed an approach to predict the relationship between two marked nouns where lexical and sentence level features are learned through a CNN. The experimental results demonstrate that the position features are critical for relation classification, while the extracted lexical and sentence level features are effective for relation classification. Zhang et al. (2015) considered that a key issue that has not been well addressed by the CNN-based method, which is the lack of capability to learn temporal features, especially long-distance dependency between nominal pairs, so they proposed a simple framework based on recurrent neural networks and compared it with CNN-based model. Experiments on two different datasets strongly indicate that the RNN-based model can deliver better performance on relation classification, and it is particularly capable of learning long-distance relation patterns. Xu et al. (2015) thought the method of Zeng et al. has been proved effectively, but it often suffers from irrelevant subsequences or clauses, especially when subjects and objects are in a longer distance. Moreover, such information will hurt the extraction performance when it is incorporated into the model. Therefore, they proposed to learn a more robust relation representation from a convolution neural network model that works on the simple dependency path between subjects and objects, which naturally characterizes the relationship between two nominal and avoids negative effects from other irrelevant chunks or clauses. Simulation results indicates that the performance has been improved to 85.4%.

3 THE PROPOSED DEEP NEURAL NETWORK

The proposed deep neural network architecture is shown in Figure 1. The deep neural network is divided into three layers. In the first layer, each character (or class key words) is mapped into character embedding (or class embedding) space. Extract features in the second layer which include:(1) Select character embedding by sliding window, then extract the character features by the neural network;(2) Character embedding of a sentence is taken as the input of the convolutional layer, and sentence features is extracted after pooling;(3) Class keywords embedding are concatenated, then input them into the neural network to extract the class features. In the last layer, we assemble the three features head-to-tail in sequence as the input of fully connected layer, to predict the label of character.

3.1 Character Embedding

Traditionally, a word is represented by a one-hot-spot vector. The vector size equals the vocabulary size. The element at the word index is “1” while the other elements are “0”s, which leads to the curse of dimensionality and sparseness. Now, a word is represented by a real-valued vector in R^M which could learn through language model.

Character embedding is a finer-grained representation of word embedding. Now, character embedding is primarily applied to Chinese segmentation via deep neural networks. The Chinese

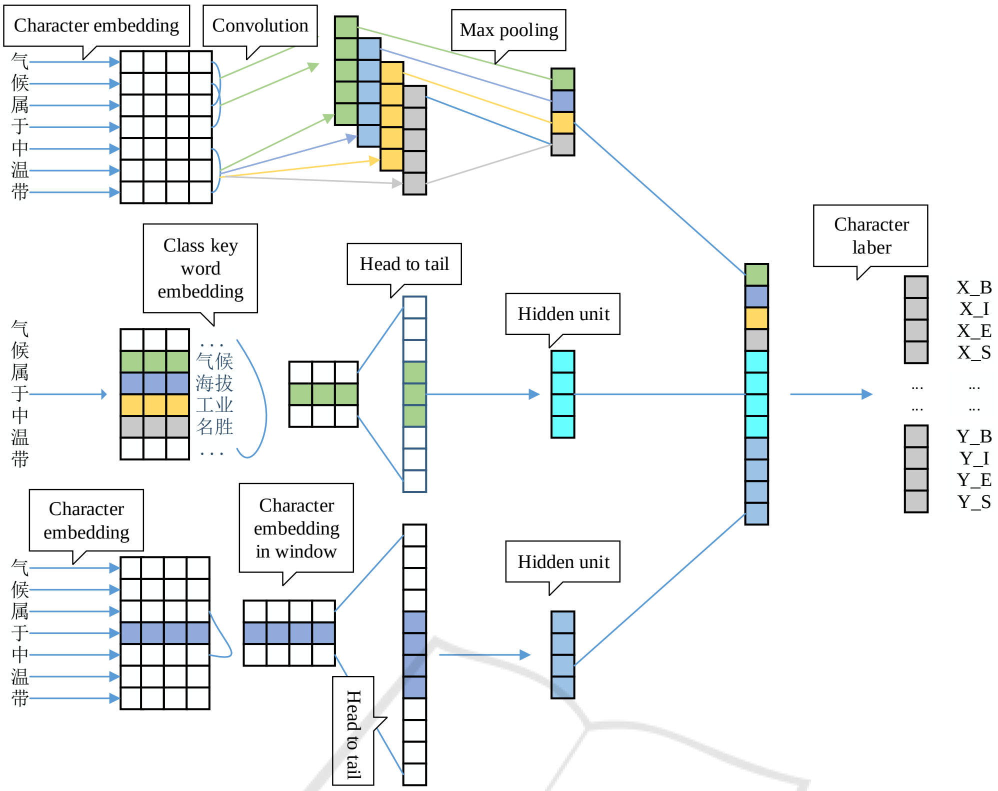


Figure 1: Architecture of the deep neural network.

segmentation, is one of the key problems in entity relation classification via deep neural networks. In these paper, we select character embedding, this is because: (1) The accuracy of ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System) is 98%, but the error propagation could be accumulated through layers, and the accuracy of Chinese segmentation in a specific area is lower than common areas. We can avoid these errors by character embedding which can be seen in Figure 2; (2) Though coarse-grained word embedding would lose important information easily in some specific sentences, finer-grained character embedding could improve the accuracy of entity relation extraction.

correct segment	阿什河/流贯/全境
ICTCLAS	阿/什/河流/贯/全境
character segment	阿/什/河/流/贯/全/境

Figure 2: segmentation of word and character.

A character is represented by a real-valued vector in R^M , where M is a hyper-parameter. A dictionary has size of D. Formally, we assume a given Chinese sentence $c_{[1:n]}$ that is a sequence of n characters c_i , $1 \le i \le n$. Then map the $c_{[1:n]}$ to a real-valued matrix $W_{[1:n]}$ that is a sequence of n character embedding W_i .

The feature vector of each character, starting from a random initialization, can be automatically trained. But the number of iterations increases, and the convergence rate is poor. We should input character embedding which contain more semantic information after training on untagged data into neural networks, which could improve the convergence speed and accuracy.

3.2 Character Feature

We tag a character or a word mainly relied on its adjacent character or word, so in this paper, we extract character feature through sliding window. We concatenate the vectors of words within a window, K is the windows size, and then we input these into the neural network to extract the features of the character. For example, in Figure 3, K=3, we extract the feature of "于" through the character representations of "属","于","中".

3.3 Sentence Feature

In some complex tasks, such as semantic role labelling and entity relation extraction, a window carrying local information is not enough, we need global information. So we combine local features of each character in a sentence into a global feature by convolution, while we enhance primary character features of a sentence by pooling. For example, in Figure 4, each character is represented by a real-valued vector in R^M. In the convolution layer, we use a convolution kernel K to extract character features around each window of the given sequence, then we perform a max pooling over them to produce a global feature vector in order to capture the most useful local features produced by the convolutional layer. We can also solve the problem that the sentence length is different.

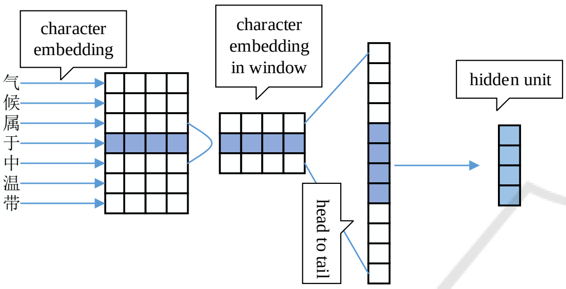


Figure 3: Character feature extraction based on neural network.

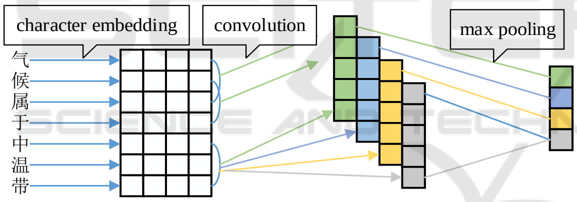


Figure 4: Sentence feature extraction via convolutional neural network.

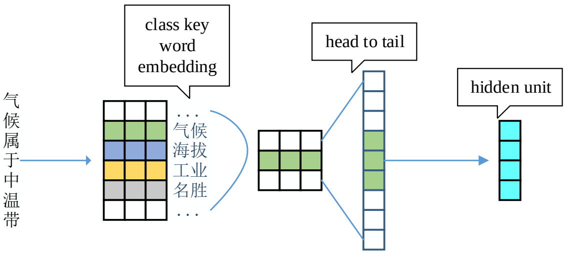


Figure 5: Class feature extraction based on neural network.

3.4 Class Feature

If we can directly extract class keywords or characters/words which can represent class keywords (some class keywords are similar to the third level attribute in Appendix 1), the feature of these characters or words could represent what this sentence want to say. And it is more directly and useful. In this paper, we call these keywords and features as class keywords and class feature. As the class keywords are independent of each other, we can extract class feature through common neural network shown in Figure 5, which uses less time than convolutional neural network.

4 DATA

As ACE relation type system and corpus aren’t suitable for geography area, we need to construct geographic entity relation type system and corpus artificially.

4.1 Relation Type

There are seven Chinese relation types in ACE-07, and the relation indicates 19 subset relationships, but geographic area is not included. We define our geographic relation type based on hierarchical model of ACE entity relation type system. Entity attribute has been divided into spatial and non-spatial attribute, while spatial attribute is divided into spatial measure, direction and shape, as well as non-spatial attribute is divided into climate, social attribute and thematic attribute. Each type is further divided into subtypes, such as length, width, altitude, annual average temperature, population, specialty, etc., as shown in appendix 1, and we focus on the third level relation type.

4.2 Entity Relation Corpus

In this paper, we get data from Chinese geography of Encyclopaedia of China and geographic terms of Baidu Baike. The text of Chinese geography of Encyclopaedia of China has following characteristics:(1) The content of each part text is mainly the description of the entity at the beginning of the text, such as ShidaoGang and ShiheziShi in Figure 6. (2) The description of the text is mainly geographical knowledge, but it also has a small amount of history, therefore we label the entity at the text beginning as the object, and other labelled as the attribute. We don’t need complex pre-processing such as Chinese word segmentation, named entity recognition, etc. What we need to do is just the ejectment of geographical irrelevant sentence and the processing of punctuation of Chinese and English. We develop seven labelling rules to annotate geographic entity relation corpus in GATE platform. The seven labelling rules are:

1. In this paper, we label the relation type of

Table 1: Annotation pattern.

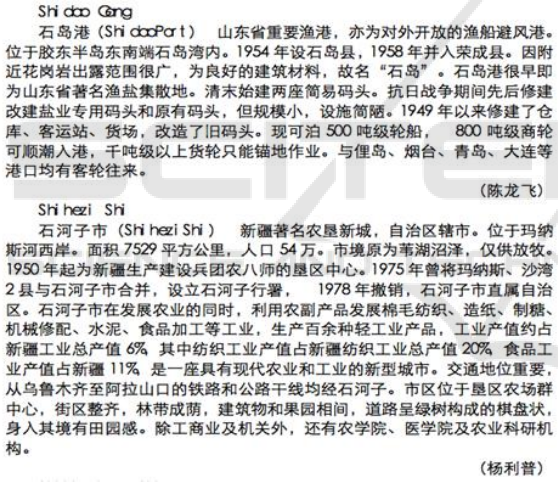
model	beginning character	middle character	ending character	single character	other
BIESS	X-B	X-I	X-E	X-S	S

Table 2: Annotation pattern sample.

面	积	1710	平	方	公	里
S	S	MJ-B	MJ-I	MJ-I	MJ-I	MJ-E

character using the following label model (Table 1), where X is the attribute name. Assuming the property name of area called MJ, "面积1710平方公里" could be labelled as in Table 2;

2. The multiple attribute value of same attribute all need to be labelled;
3. Don't need to label the said range of character or word;
4. "~" need to be labelled;
5. Only label the place name for the highway and railway;
6. Generalize the type;
7. Character in brackets are labelled as S.



Shi dao Gang

石岛港（Shi daoPort） 山东省重要渔港，亦为对外开放的渔船避风港。位于胶东半岛东南端石岛湾内。1954 年设石岛县，1958 年并入荣成县。因附近花岗岩出露范围很广，为良好的建筑材料，故名"石岛"。石岛港很早即为山东省著名渔盐集散地。清末始建两座简易码头。抗日战争期间先后修建改建盐业专用码头和原有码头，但规模小，设施简陋。1949 年以来修建了仓库、客运站、货场，改造了旧码头。现可泊 500 吨级轮船， 800 吨级商轮可顺潮入港，千吨级以上货轮只能锚地作业。与俚岛、烟台、青岛、大连等港口均有客轮往来。

（陈龙飞）

Shi hezi Shi

石河子市（Shi hezi Shi） 新疆著名农垦新城，自治区辖市。位于玛纳斯河西岸。面积 7529 平方公里，人口 54 万。市境原为苇湖沼泽，仅供放牧。1950 年起为新疆生产建设兵团农八师的垦区中心。1975 年曾将玛纳斯、沙湾 2 县与石河子市合并，设立石河子行署， 1978 年撤销，石河子市直属自治区。石河子市在发展农业的同时，利用农副产品发展棉毛纺织、造纸、制糖、机械修配、水泥、食品加工等工业，生产百余种轻工业产品，工业产值约占新疆工业总产值 6% 其中纺织工业产值占新疆纺织工业总产值 20% 食品工业产值占新疆 11% 是一座具有现代农业和工业的新型城市。交通地位重要，从乌鲁木齐至阿拉山口的铁路和公路干线均经石河子。市区位于垦区农场群中心，街区整齐，林带成荫，建筑物和果园相间，道路呈绿树构成的棋盘状，身入其境有田园感。除工商业及机关外，还有农学院、医学院及农业科研机构。

（杨利普）

Figure 6: The entity description in Chinese geography of Encyclopedia of China.

5 EXPERIMENTAL EVALUATION

5.1 Data Sets

In the experiment, we choose 80% of the corpus as the training set and the others as the test set as shown in the Table 3. And we conduct experiments use Keras-0.3.0 which is a deep learning framework with CPU+GPU (12+2880CORES).

Table 3: Experiment data.

Data set	Size(characters)
Corpus	37,431
Training set	29,945
Test set	7,486

5.2 The Choice of Hyper-parameters

We need to select the dimension of the character embedding and the class keyword embedding. Furthermore, the deep learning framework requires the specification of multiple hyper-parameters. This includes the size k of the character window, the size h of the hidden layers, the learning rate λ, and the dropout rate. In this paper, we adjust these hyper-parameters and consult the value range of Yanjun Qi et al. (2014), Xiaoqing Zheng (2013) and Xinchi Chen (2015) in Table 4.

F-scores on the same data set versus window size, character embedding size, class keyword embedding size, number of hidden units, dropout rate, learning rate, are shown in Figure 7. It can be seen that the F-scores drop when the window size is larger than nine which shows that the size of windows is too large so the information will be redundant. However, if the window size is too small, then the information will be deficient. Generally, the number of the hidden units has a limited impact on the performance if it is large enough. Because of the limit size of corpus that we used, we can see that the performance is no longer increasing when the hidden number is larger than 600. Dropout is an effective tool in prevent over-fitting, it is helpful to improve performance when the dropout rate is 0.2.

We also adjust the learning rate consulting the value range of Collobert et al (2011), in our experiment. Results indicate that F-score is highest when learning rate is 0.015. The hyper-parameters we selected are shown in Table 4.

Table 4: Hyper-parameters.

Hyper-parameters	Value range	Selected value
Sliding window	3,5,7,9,11,13,15	9
Character embedding size	50,80,100,120,150,200	100
Class keyword embedding size	100,200,300,400,500,600	300
Number of hidden units	300,500,600,700,900	600
Dropout rate	0,0.2,0.5	0.2
Learning rate	0.005,0.01,0.015,0.02	0.015

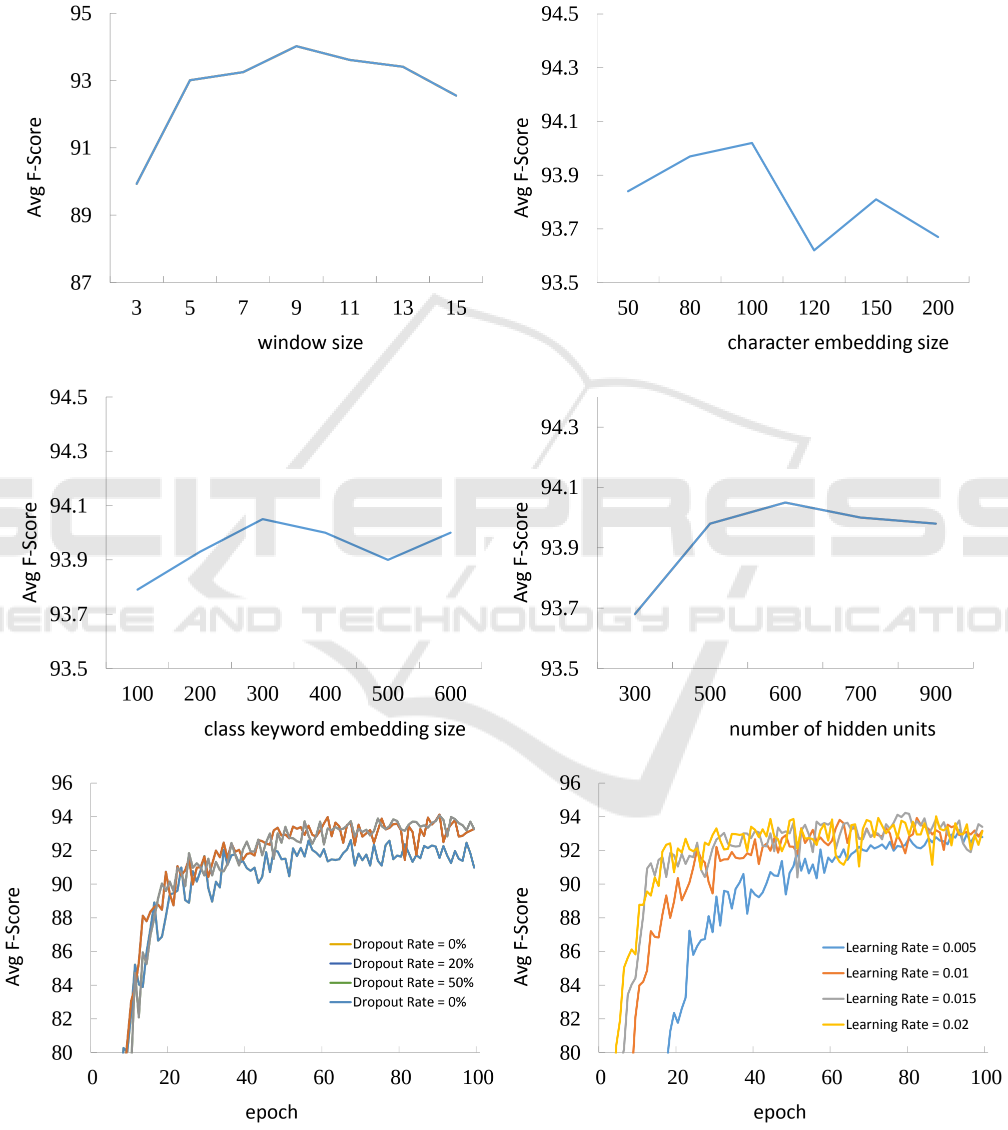


Figure 7: Average F-scores versus different hyper-parameters.

Table 5: Experiment results based on the random initialization character vector.

configuration	P_W	P_{Avg}	R_{Avg}	F_{Avg}	Time(s)
CC	94.35	93.47	83.96	88.13	590
CC+SC	94.84	94.48	85.31	89.4	40648
CC+TC	96.12	96.33	90.57	**93.19**	874
CC+SC+TC	96.39	96.48	91.51	**93.74**	40582

Table 6: Experiment results based on word2vec character embedding.

configuration	P_W	P_R	P_{Avg}	R_{Avg}	F_{Avg}	Time(s)
W2V+CC	94.86	89.78	93.35	85.85	89.15	459
W2V+CC+SC	95.71	92.64	95.06	88.95	91.65	42539
W2V+CC+TC	96.35	94.94	96.96	92.32	**94.36**	707
W2V+CC+SC+TC	96.69	95.41	96.54	92.99	**94.56**	40550

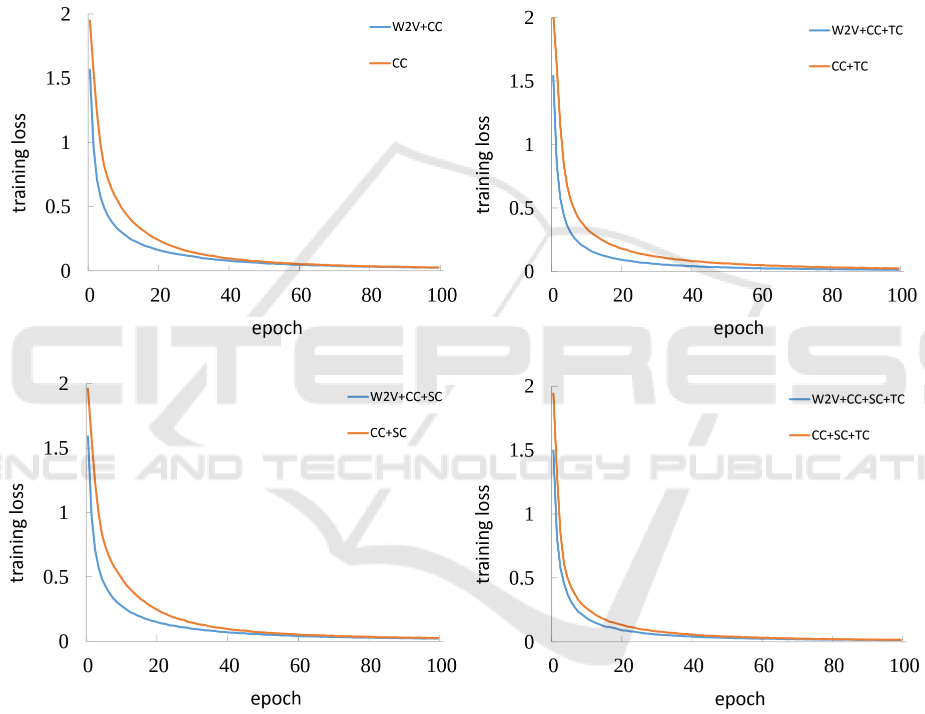


Figure 8: The influence of word2vec character embedding on the convergence speed.

5.3 Results and Discussion

We systematically test the incremental combinations of multiple characters. For convenience, we rename Word2Vec character embedding as W2V, character feature as CC, sentence feature as SC, class feature as TC.

At first, we conduct contrast experiments on different combination of the features based on the random initialization character vector. In the supervised methods setting, relation extraction is expressed as a classification task (tagging characters). Hence, metrics like Precision (P), Recall (R) and F-Measure (F) are used for performance evaluation. Moreover, we also choose the weighted average of all kind of metrics : F_{Avg},P_{Avg}, R_{Avg}.

$$F_{Avg} = \frac{\sum_{i=1}^{n}(C_i * F_i)}{\sum_{i=1}^{n} C_i} \quad (1)$$

$$P_{Avg} = \frac{\sum_{i=1}^{n}(C_i * P_i)}{\sum_{i=1}^{n} C_i} \quad (2)$$

$$R_{Avg} = \frac{\sum_{i=1}^{n}(C_i * R_i)}{\sum_{i=1}^{n} C_i} \quad (3)$$

Where, C_i is the number of i^{th} class. And we add accuracy of character tagging (P_W) and accuracy of relation character tagging (P_R).

$$P_W = \frac{N_c}{N_{ct}} \tag{4}$$

$$P_R = \frac{N_{rc}}{N_{rct}} \tag{5}$$

Where, N_c is the number of correctly tagging character, N_{ct} is the number of tagging character in the test data set, N_{rc} is the number of correctly tagging relation character, N_{rct} is the number of relation character in the test data set.

As shown in Table 5, it is clear that the performance has improved by adding sentence features or class features, which makes up the error of recognition because of the lack of window information in long or complex sentences. It is obvious that the performance of combination of CC+TC has enhanced greatly. We need to design class keywords manually, but training time is reduced dramatically. As shown in Table 5, the time of CC+TC (874s) is 1/48 of CC+SC (40648s) has reduced about 4%. What if we could not find class keywords in sentences? In that case, we need sentence features, as in Table 5, we can see that $\mathrm{F}_{\mathrm{Avg}}$ of CC+SC+TC is higher than CC+TC.

When we compare CC, CC+SC and CC+TC, we can find that CC+SC only brings an improvement of 1.3%, lower than CC+TC. There are three reasons. Firstly, the information that sliding window (size is 9) contained is abundant for the label of characters. Secondly, the class features are extracted from class keywords which represent the global features of sentence. Sentence features are extracted from characters which only represent local features of sentence, therefore, under certain conditions, class features are more effective than sentence features. Thirdly, as information redundancy of sentences, we extract sentence features via convolutional neural networks rely on big data.

Xiaoqing (2013), Wenzhe (2014) and other scholars found that the pre-training character embedding could improve the accuracy of segmentation in experiment of Chinese word segmentation. We train the character embedding by Word2Vec of Mikolov (2013). In this experiment, we find that the $\mathrm{F}_{\mathrm{Avg}}$ of the four combinations has improved as shown in Table 6. It is clear that $\mathrm{F}_{\mathrm{Avg}}$ of W2V+CC+SC+TC is highest. Another advantage of character embedding trained by Word2Vct is fast convergent. As shown in Figure 8, we can see effect of W2V on convergence speed. Since the experimental purpose and data are not the same, we do not compare our method with other published methods.

6 CONCLUSION AND FUTURE WORK

In this work, we firstly construct geographic entity relation type system and corpus, and then we make our data public freely on the web for others. Secondly, we tackle the Chinese geographical knowledge entity relation extraction task using a deep neural network. We perform the experiments using the geographic dataset we constructed. Our experimental results show that we achieve a good performance with the averaged overall precision of 96.54%, averaged recall of 92.99%, and averaged F value of 94.56%, which are relatively high.

In future, we will explore the following research directions: (1) This paper only considers the triples relations, which are not effective for some complex entity relations, such as “AkesuShi is called HuiCheng in 1757 and it is called HanCheng in 1883.”; (2) The training data is narrower, we need more extensive corpus; (3) The hierarchy of conception in this paper is relatively simple, we need to complex hierarchy concept.

ACKNOWLEDGEMENTS

This work was supported in part by the National High-tech R&D Program of China (863 Program) under Grant No.2015AA015403, Science & Technology Pillar Program of Hubei Province under Grant No.2014BAA146, Nature Science Foundation of Hubei Province under Grant No.2015CFA059, Science and Technology Open Cooperation Program of Henan Province under Grant No.152106000048, the Fundamental Research Funds for the Central Universities under Grant No.2016-zy-047, CERNET Innovation Project under Grant No. NGII20150309.

REFERENCES

Santos, C., & Zadrozny, B. 2014. Learning Character-level Representations for Part-of-Speech Tagging. *ICML* 1818-1826.

Kim, Y. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv*, 1408(5882).

Santos, C., & Gatti, M. 2014. Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts. *COLING*, 69-78.

Kalchbrenner, N., et al., 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv*,1404(2188).

Collobert, R., et al., 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 2493-2537.

Hu, B., et al., 2014. Convolutional neural network architectures for matching natural language sentences. *Advances in Neural Information Processing Systems*, 2042-2050.

LeCun, Y., et al., 2015. Deep learning. *Nature*,521(7553), 436-444.

Socher, R., et al., 2012. Semantic compositionality through recursive matrix-vector spaces. *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 1201-1211.

Mikolov, T., et al., 2013. Linguistic Regularities in Continuous Space Word Representations. *HLT-NAACL* ,746-751.

Zeng, D., et al., 2014. Relation Classification via Convolutional Deep Neural Network. *COLING*, 2335-2344.

Zhang, D., & Wang, D. 2015. Relation Classification via Recurrent Neural Network. *arXiv preprint arXiv*, 1508(01006).

Xu, K., et al., 2015. Semantic relation classification via convolutional neural networks with simple negative sampling. *arXiv preprint arXiv*, 1506(07650).

Qi, Y., Das, et al., 2014. Deep learning for character-based information extraction. *European Conference on Information Retrieval*, 668-674.

Zheng, X., et al., 2013. Deep Learning for Chinese Word Segmentation and POS Tagging. *EMNLP*, 647-657.

Chen, X., et al., 2015. Long short-term memory neural networks for chinese word segmentation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*,1385-1394.

Pei, W., et al., 2014. Max-Margin Tensor Neural Network for Chinese Word Segmentation. *ACL*, 293-303.

Mikolov, T., et al., 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv*,1301(3781)

APPENDIX

Geographical entity relation type system.

first level attribute	second level attribute	third level attribute
spatial attribute	spatial measure	length
		width
		altitude
		area
		east longitude
		west longitude
		south latitude
		north latitude
		location
	direction	orientation
	shape	terrain landform(geological structure)
		terrain
non-spatial attribute	climate	annual mean temperature
		monthly mean temperature
		mean temperature in Jan
		mean temperature in July
		the lowest temperature
		the highest temperature
		annual precipitation
		average annual sunshine
		rainy days
		frost free days
		climate
	social attribute	population
		provincial capital
		county(county resident)
		city(city resident)
		country
		province
		city
		nickname
	thematic attribute	mineral
		agriculture and husbandry
		industry
		specialty
		river
		highway
		railway
		places of interest（tour）