Chinese Hedge Scope Detection Based on Phrase Semantic Representation

Huiwei Zhou, Shixian Ning, Yunlong Yang, Zhuang Liu and Junli Xu
School of Computer Science and Technology
Dalian University of Technology
Dalian, China
zhouhuiwei@dlut.edu.cn; {ningshixian, SDyyl_1949, zhuangliu1992, xjlhello}@mail.dlut.edu.cn

***Abstract*—Chinese hedge scope detection is dependent on syntactic and semantic information. Most previous methods typically use lexical and syntactic information as a basic unit of classification, which make these methods lose part of the effective structure information. In order to enhance detection performance, we take the phrase, which are extracted from the parse tree by some heuristic rules, as the classification unit (candidate phrase). Furthermore, a novel hierarchical neural network is proposed to learn the semantic representation of the phrase and its context. Experiments on the Chinese Biomedical Hedge Information (CBHI) corpus show that our system could achieve state-of-the-art performance without using any complicated feature engineering.**

Keywords-Chinese hedge scope detection; phrase semantic representation; hierarchical neural network

I. Introduction

Hedge is prevalent both in English and Chinese, and received widespread attention in the NLP community [1]. Hedge scope detection aims to identify which tokens are affected by hedge. For example, “研究报道 RECS1 **可能** 位于细胞的内核体. (Studies have shown that RECS1 **may** be located in the nucleus of the cell.)”, “可能 (may)” is a hedge cue and its scope is the statement “RECS1 可能位于细胞的内核体 (RECS1 may be located in the nucleus of the cell)”, while leaving the other tokens outside.

Hedge scope detection is of great significance for information extraction in the Biomedical domain. The CoNLL-2010 shared task on automatically detecting the in-sentence scope of hedge cues [2], has greatly contributed to the research of English hedge information detection.

Generally, hedge scope detection approaches can be divided into two categories: rule based methods and machine learning based methods. Rule-based methods [3] aim at finding and extracting the constraints and the relationships between hedge and their control scope. They are simple and effective, but lack flexibility and universality.

Machine learning-based methods regard the task as a token classification problem, which usually classify each token in a sentence as being the first element of the scope (F-scope), the last (L-scope), or neither (None) [4]. Machine learning-based methods mainly include two methods: feature-based methods and kernel-based methods. Feature-based methods [5-6] design a set of discrete features with “one-hot” representations based on lexical and flat syntactic information. Tree kernel-based method [7] capture structured syntactic information by counting the number of common sub-trees. Machine learning-based methods could merely capture shallow lexical information and syntactic information for hedge scope detection, and rely on complex feature engineering.

Recent years, deep learning have been widely used in the field of NLP. It can automatically construct the semantic representation through the multi-layer neural network, such as Convolutional Neural Network (CNN) [8] and Long Short Term Memory networks (LSTM) [9]. Fancellu et al. [10] use the word embedding and POS features to construct an English hedge scope detection system based on Bidirectional LSTM. Qian et al. [11] extract the features from various syntactic paths between the hedge and its candidate in both constituency and dependency parse trees based on CNN. Zhou et al. [1] incorporate flat information, syntactic structure information and semantic information for Chinese hedge scope detection based on the fusion of composite kernel and LSTM.

However, these methods regard the word as the classification unit and could not fully take advantage of the compositional meaning of the phrase in parse tree, and therefore prevent a deeper understanding of the hedge scope.

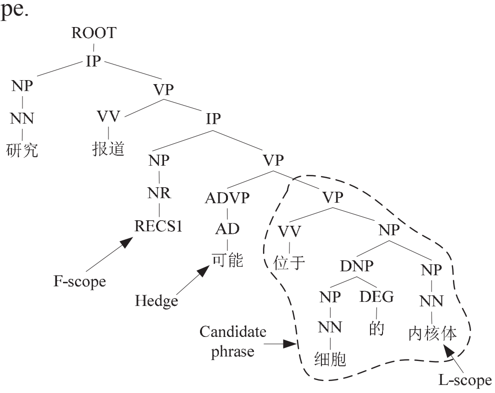

Figure 1. Phrase parse tree.

In order to remedy this defect, we propose a Phrase Encoded Hierarchical Neural Network (PHNN). Firstly, we use Stanford Parser to get the phrase parse tree of the sentence. The phrase parse tree of sentence 1 is shown in Fig. 1. Inspired by Zhu et al. [12], we use the following heuristic rule to extract the candidate phrase. Except for the hedge and its ancestors, all the phrases which parent constituent cross-bracket or include the hedge cue are selected as candidate boundary phrases for Chinese hedge scope detection. In Fig. 1, since the parent constituent of the VP phrase “位于细胞的内核体 (located in the nucleus of the cell)” contains the hedge cue “可能 (may)”, so it was selected as a candidate phrase.

978-1-5386-1981-0/17/$31.00 ©2017 IEEE

Then, an LSTM model is used to learn the semantic representation of the candidate phrase, which could take the order of word into account. Finally, the context words of hedge and its candidate phrase representation are jointed as a sequence fed to a CNN, which could capture the deep context semantic representation of hedge and its candidate phrase.

II. Methods

A. Data Preprocessing

In this section, we present the data preprocessing for the acquisition of candidate samples. First of all, each word is represented as a *d*-dimensional word embedding $w_i \in R^{d_1}$ (d_1=100). Then, the candidate phrase with a length of *n* is defined as $x_p = \{w_1, ..., w_i, ..., w_n\}$ and its context in the window [−2, 2] is defined as $x_{candidate} = \{w_{-2}, w_{-1}, x_p, w_{n+1}, w_{n+2}\}$. The same operation is performed to the hedge cue w_c and thus we can get its context sequence $x_{cue} = \{w_{c-2}, w_{c-1}, w_c, w_{c+1}, w_{c+2}\}$.

In addition, the constituent (such as NP, VP) representation of candidate phrase $f_p \in R^{d_2}$ (d_2=10) is concatenated to each word representation of $x_{candidate}$, and the POS representation of hedge $f_c \in R^{d_2}$ is concatenated to each word representation of x_{cue}.

Finally, we concatenate the two context sequences to form the final training sample. Take the left boundary as an example, the two context sequences are concatenated as a candidate sample $x = \{x_{candidate}; f_p \,||\, x_{cue}; f_c\}$ (|| means concatenation) and then fed to our hierarchical neural network.

B. Phrase Encoded Hierarchical Neural Network

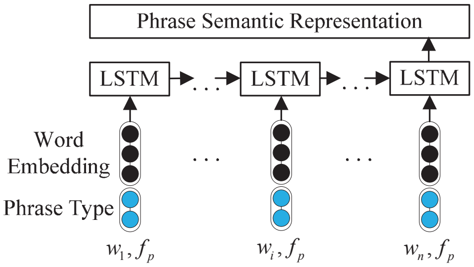

Figure 2. The architecture of phrase semantic representation learning.

1) Phrase Semantic Representation Learning: The learning process is shown in the Fig. 2. An LSTM layer introduces a memory cell to preserve states over long periods of time, and controls the update of hidden state and memory cell by three types of gates, forget gate, input gate, and output gate respectively. Let *W* and *U* be the transfer matrix, *b* be a bias term, *h* represents the hidden state representation, *c* represents memory cell, σ be a sigmoid activation function and $\odot$ denote component-wise multiplication. Concretely, each step of LSTM unit is shown in the following formula:

$$f_t = \sigma(W^{(f)} x_p + U^{(f)} h_{t-1} + b^{(f)}) \tag{1}$$

$$i_t = \sigma(W^{(i)} x_p + U^{(i)} h_{t-1} + b^{(i)}) \tag{2}$$

$$o_t = \sigma(W^{(o)} x_p + U^{(o)} h_{t-1} + b^{(o)}) \tag{3}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W^{(c)} x_p + U^{(c)} h_{t-1} + b^{(c)}) \tag{4}$$

$$h_t = o_t \odot tanh(c_t) \tag{5}$$

In our work, we treat the hidden representation at the last time step as the final phrase semantic representation, and then we can get a new candidate sample $x = \{w_{-2}; f_p, w_{-1}; f_p, h_t, w_{n+1}; f_p, w_{n+2}; f_p \,||\, x_{cue}; f_c\}$.

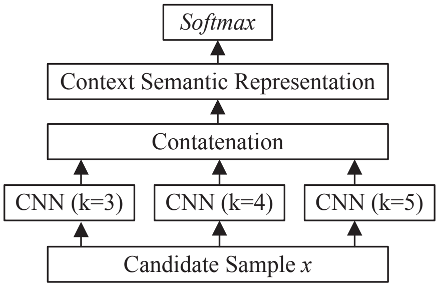

Figure 3. The architecture of context semantic representation learning.

2) Context Semantic Representation Learning: The learning process is shown in the Fig. 3. A CNN model is mainly composed of convolutional layer and pooling layer. In convolutional layer, we slide multiple filters *w* over the candidate sample *x* and compute the dot product to obtain different feature maps. The convolution operation with a filter window size *k* can be expressed as $featuremaps_i = f\left(w \bullet x_{i:i+k-1} + b\right)$, here *b* is a bias term and *f* is a nonlinear function. In practice, multiple filters with different widths *k* ($k = 3, 4, 5$) are used to further obtain a set of different feature maps. Then we use a max pooling operation to take the largest element from the rectified feature map. Finally, we concatenate these features for classifying the candidate sample into various classes through a *softmax* activation function.

C. Postprocessing

Hedge scope is a continuous sequence of tokens. However, the classifier does not guarantee that only one left boundary and one right boundary are obtained for one cue. So we apply the following rules to post-process the classification results.

- If one token is predicted as F-scope and one token as L-scope, the sequence will start at the token predicted as F-scope, and end at the token predicted as L-scope.
- If one token is predicted as F-scope, and none or more than one token is predicted as L-scope, the sequence will start at the token predicted as F-scope, and end at the token with the maximum L-scope predicted result.
- If one token is predicted as L-scope, and none or more than one token is predicted as F-scope, the sequence will start at the token with the maximum F-scope predicted result, and end at the token predicted as L-scope.

TABLE I. COMPARISON OF WORD-BASED METHOD VS. PHRASE-BASED METHOD

Method	Feature	F-scope			L-scope			*F*1_sen(%)
		P (%)	*R (%)*	*F (%)*	*P (%)*	*R (%)*	*F (%)*	
Word-based	None	48.64	62.93	54.87	59.77	77.01	67.30	52.05
	+POS	72.69	83.04	63.50	69.57	74.03	71.73	57.33
	+PhraseType	68.15	62.19	65.03	72.69	83.04	77.52	59.57
Phrase-based	None	56.38	62.24	59.16	69.64	81.71	75.19	56.59
	+POS	61.69	57.12	59.32	73.73	79.95	76.71	58.13
	+PhraseType	70.78	65.49	**68.03**	79.47	86.93	**83.04**	**63.52**

TABLE II. COMPARISON OF DIFFERENT PHRASE REPRESENTATION LEARNING METHODS

Method		F-scope			L-scope			*F*1_sen(%)
		P (%)	*R (%)*	*F (%)*	*P (%)*	*R (%)*	*F (%)*	
Vector Composition	Point-wise Addition	74.76	62.40	68.02	82.75	79.57	81.13	64.91
	Point-wise Average	71.20	63.68	67.23	83.51	80.27	81.86	64.27
Neural Network	CNN	72.39	75.52	73.92	80.25	87.36	83.66	66.24
	LSTM	72.20	76.75	**74.41**	82.07	88.64	**85.23**	**67.84**
	Bidirectional LSTM	73.39	72.21	72.80	87.91	80.69	84.15	67.52
	LSTM+CNN	69.52	78.93	73.93	81.12	87.52	84.20	67.31

III. EXPERIMENTS

Experiments are conducted on the Chinese Biomedical Hedge Information (CBHI) corpus [13], which contains a total of 9385 sentences. We randomly selected 7510 sentences for training and 1875 sentences for testing. We evaluated our models in terms of precision (*P*), recall (*R*), and *F*1-score (*F*). Furthermore, in order to assess the combined result after post-processing operation, we also introduced *F*1-score on sentence-level (*F*1_sen) with the following form:

$$F1\_sen = \frac{\#total\_correct}{\#total\_test} \quad (6)$$

where $\#total\_correct$ is the correct number of sentences, and $\#total\_test$ is the total number of sentences in the test set. $F1\_sen$ takes the whole sentence as a basic unit to carry out evaluation and determine whether the predict results and the correct results are exactly the same.

A. Training and Implementation Details

Word embeddings are pre-trained on SogouCS corpus[1] (1.65GB) with Word2Vec [2] toolkit. 110 filters with window size $k = 3,4,5$ respectively are used in CNN. The dimension of the hidden layer in LSTM is also 110.

Nesterov Adam optimizer is used and its initial learning rate is set to 0.002. Mini-batch size is 128. In order to alleviate the over-fitting problem, we apply the dropout method on both the input and output of LSTM layer. We fix dropout rate at 0.2 throughout the experiments. ReLU activations are also used to help in speeding up the training and avoid neuron saturation.

B. Word-based Method vs. Phrase-based Method

Tab. I lists the results of the Word-based method and the Phrase-based method respectively. The Word-based method take the leftmost (rightmost) word of candidate phrase as classification unit; the Phrase-based method take the whole candidate phrase as classification units. Both of the two methods only adopt a single CNN architecture.

From the table we can see, the performance of the Phrase-based method is significantly better than the Word-based method. It shows that phrases could provide more information to the hedge scope detection task. Meanwhile, with the integration of additional features, detection performance also improved continuously. Since that all of the lexical features are effective for scope detection.

C. Comparison of Different Phrase Semantic Representation Learning Methods

In this section, we investigate our PHNN model with different phrase semantic representation learning methods. The details are shown in Tab. II. Among them, LSTM+CNN method means that LSTM and CNN are used to learn phrase semantic representation respectively, and then concatenate the results of both. From Tab. II, we know that:

1) The results of vector composition methods are worse than neural network methods. This is because the former could only capture the shallow semantic information, while ignoring the context dependent information within the candidate phrase. On the contrary, the latter could make full use of the internal semantic dependent information within the phrase, thus improves the performance of hedge scope detection.

2) In the neural network method, the result of CNN is worse than other models. This may be because LSTM is suitable for modeling the continuous phrase sequences with strong temporal dependency, and characterizing context dependency of phrase effectively, therefore resulting in higher quality phrase semantic representation. Among all the methods, LSTM achieved the best performance.

[1] Available at http://www.sogou.com/labs/resource/cs.php.

[2] Available at https://code.google.com/p/word2vec/.

TABLE III. SYSTEM PERFORMANCE ON VARIOUS WINDOW SIZE COMBINATIONS

Feature	F-scope			L-scope			*F*1_sen(%)
	P (%)	*R (%)*	*F (%)*	*P (%)*	*R (%)*	*F (%)*	
PHNN (*k*=3)	72.20	76.75	74.41	82.07	88.64	85.23	67.84
PHNN (*k*=4)	71.04	76.53	73.68	84.04	86.77	85.38	67.15
PHNN (*k*=5)	75.39	72.85	74.10	84.04	86.77	85.38	67.95
PHNN (*k*=3,4)	76.37	72.91	74.60	86.43	85.23	85.82	70.19
PHNN (*k*=3,5)	76.09	71.47	73.71	82.20	90.88	86.32	69.55
PHNN (*k*=3,4,5)	73.93	75.47	74.69	83.20	87.41	85.25	**70.40**

D. Evaluation of Different CNN Filter Widths

We investigate the effects of different window sizes of filters by running the proposed PHNN model on window size *k* of 3, 4 and 5 respectively. In order to further understand the effects of the model on multiple window sizes, we test it on the extra combinations of window size, ($k = 3,4$), ($k = 3,5$) and ($k = 3,4,5$).

From Tab. III, we found that it is unclear which window size is the best size for PHNN on Chinese hedge scope detection. But compared with single window sizes, the results for multiple window sizes are consistently enhanced when more window sizes are included, resulting in the best performance when all the window sizes 3, 4 and 5 are employed. This demonstrates the advantages of the models with multiple window sizes over the single window size models. To a certain extent, it further reflects the combination of context semantic information, which are obtained by different local perceptual domains, contributes to the judgment of the scope boundary.

E. Comparison with related works

Tab. IV compare the result of our model with Zhou et al. [1]. Their best *F*1_sen score 69.92% is obtained by a hybrid system, which adopts the composite kernel model and the LSTM model to capture lexical, syntactic and semantic information.

TABLE IV. CONTRAST WITH RELATED WORKS

Model	F-scope	L-scope	*F*1_sen(%)
Zhou et al. [1]	-	-	69.92
Our work	74.69	85.25	**70.40**

As we can see, our model achieves 70.40% *F*1_sen score, which is superior to Zhou et al. [1] without complex feature engineering. We attribute our performance advantage to the phrase semantic representation learning and different n-gram context semantic features learning.

IV. CONCLUSION

In this paper, we focus on Chinese hedge scope detection and propose a hierarchical neural network PHNN, which contains an LSTM module for phrase semantic representation learning and a CNN module for context semantic representation learning. We regard phrases as the basic classification unit and learn its representation specially. Experiments on the CBHI corpus show the effectiveness of our work and it achieves 70.40% *F*1_sen.

For the future work, we will investigate other phrase representation and context representation learning methods, to further improve the performance of Chinese hedge scope detection.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China (No. 61772109, No. 61272375) and the Ministry of education of Humanities and Social Science project (No. 17YJA740076).

REFERENCES

[1] H. W. Zhou, J. L. Xu, Y. L. Yang, H. J. Deng, L. Chen, D. G. Huang, “Chinese hedge scope detection based on structure and semantic information,” in China National Conference on Chinese Computational Linguistics. CCL, 2016, pp. 204-215.

[2] R. Farkas, V. Vincze, G. Móra, J. Csirik, G. Szarvas, “The CoNLL-2010 shared task: learning to detect hedges and their scope in natural language text,” in Proceedings of the Fourteenth Conference on Computational Natural Language Learning. ACL, 2010, pp. 1–12.

[3] A. Özgür, D. R. Radev, “Detecting speculations and their scopes in scientific text,” in Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. ACL, pp. 1398–1407.

[4] H. W. Zhou, H. J. Deng, D. G. Huang, “Hedge scope detection in biomedical texts: An effective dependency-based method,” PloS one, vol. 10, no. 7, pp. e0133715, 2015.

[5] R. Morante, V. V. Asch, W. Daelemans, “Memory-based resolution of in-sentence scopes of hedge cues,” in Proceedings of the Fourteenth Conference on Computational Natural Language Learning. ACL, 2010, pp. 40–47.

[6] B. W. Zou, Q. M. Zhu, G. D. Zhou, “Negation and speculation identification in Chinese language,” in Proceedings of ACL-IJCNLP. ACL, 2015, pp. 656–665.

[7] B. W. Zou, G. D. Zhou, Q. M. Zhu, “Tree kernel-based negation and speculation scope detection with structured syntactic parse features,” in Proceedings of EMNLP. ACL, 2013, pp. 968–976.

[8] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.

[9] S. Hochreiter, J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, pp. 1735-1780, 1997

[10] F. Fancellu, A. Lopez, B. L. Webber, “Neural networks for negation scope detection,” in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. ACL, 2016, pp. 495-504.

[11] Z. Qian, P. Li, Q. Zhu, G. Zhou, Z. Luo, W. Luo, “Speculation and negation scope detection via convolutional neural networks,” in Proceedings of EMNLP. ACL, 2016, pp. 815-825.

[12] Q. M. Zhu, J. H. Li, H. L. Wang, G. D. Zhou, “A unified framework for scope learning via simplified shallow semantic parsing,” in Proceedings of EMNLP. ACL, 2016, pp. 714–724.

[13] H. W. Zhou, H. Yang, J. Zhang, S. Y. Kang, D. G. Huang, “The research and construction of Chinese hedge corpus,” Journal of Chinese Information Processing, vol. 29, pp. 83–89, 2015.