Automatic Identification of Chinese Prepositional Phrase Including Verbs

Yuanyuan Wang and Jianliang Xu

Abstract—**In this paper, with the help of corpus which was changed by the Harbin Institute of Technology Information Retrieval Laboratory's Chinese dependency-based Treebank and by using the parser MaltParser as an implementation tool, this paper proposed and implementation a Chinese dependency parsing algorithm, in order to effectively identify the right boundary of the prepositional phrases that contains a verb. Empirical results show that this algorithm has improved the error when analyze the prepositional phrase containing a verb.**

Index Terms—**Dependency parsing, parsing algorithm, prepositional phrase, right-boundary**

I. Introduction

Natural language processing has become a hot research in recent years, and syntax analysis is one of the key technologies in it. The main task of Syntactic analysis is finding the relationship between words and phrases in the sentences and how they construct the sentences. Prepositional phrases occupy a significant proportion in many Chinese phrase types. By the statistics of extracting 26551 sentences from the 2000 the people daily corpus, J.W.Gan found that there are 15.1 appeared in the prepositional phrase [1] on the average of every 100 words. Processing prepositional phrases correctly and effectively in Chinese sentences plays a significant role in the syntactic analysis.

The internal structure of the prepositional phrase is very complex, but it has a remarkable characteristic of boundary. Hence, the main task is to find a prepositional phrase boundary in Chinese syntactic analysis. PP2 Attachment problem is the main research in the analysis of English prepositional phrases[2]-[3]. In the Chinese prepositional phrase research, generally researchers do the process by putting the prepositional phrases into syntactic analysis, which used HMM[4]-[5],MBL[6],the maximum entropy[7]-[8], SVM[9] method to research block analysis, basic phrase recognition and then have achieved satisfactory results.

In this paper, we also focus our interest on syntactic analysis processing. Different from the Arc-eager algorithm [10] and BPP algorithm [11], we however explore the problem of identifying the right boundary of the prepositional phrase which contains a verb.

II. Parsing Algorithm

The main research in the Chinese prepositional phrase is to find the boundary of the Prepositional phrases which means to find the start position and the end position of the prepositional phrases. The left boundary of the prepositional phrase is the preposition itself, so identifying the right boundary of prepositional phrases correctly is the key work. Many previous algorithms on syntactic analysis have not resolved the problem of identifying the right boundary of the prepositional phrases effectively which contains a verb, such as Arc-eager algorithm[10] and BPP algorithm[11].

Arc-eager algorithm: Arc-eager algorithm always compares the top word of the stack with the next input word. It is just effective to the short distance dependence analysis, but error occurs when dealing with long distance dependence analysis. For instance, 'The story happened in Shandong's small town'. After the Arc-eager algorithm's analysis, the dependent of preposition 'in' is 'Shandong', and in fact the true dependence is 'town'. This is because the Arc-eager algorithm always compares stack top word with the next input word. The preposition 'in' first determines the interdependent relationship with 'Shandong', and then lost it with 'town'. The word 'town' cannot find the correct head 'in' but find the wrong head 'happened'. That's the usual error of the Arc-eager algorithm in processing prepositional phrase's long distance dependency.

BPP algorithm: Compared with Arc-eager algorithm's analysis, BPP algorithm first identifies the prepositional phrase and then uses the same algorithm to analyze the sentence. And then let the heart word instead of the entire phrases to involve in the analysis of sentence. The BPP algorithm pointed that the left margin of the prepositional phrase is preposition itself. The right boundary of the prepositional phrase always followed by the word which has the characteristics of 'de', punctuation, verbs, prepositions and adverbs. If you find a preposition when doing syntactic analysis, and there is a word of the preposition behind match the above characteristics, then the prefix of the word is the right sector of the preposition. But if the prepositional phrase contains a verb, the BPP algorithm appears the errors. Therefore, in this paper we propose this algorithm to solve boundary identification problem of the prepositional phrase which contains a verb.

III. Algoritnm Analysis

A. Algorithm Description

The proposed algorithm includes seven points:

Manuscript received June 25, 2012; revised July 24, 2012.

The authors are with the Department of Computer Sciences, Ocean University, Qingdao, 266100, China (e-mail: oucwyy@gmail.com, oucwyy@gmail.com)

DOI: 10.7763/IJFCC.2012.V2.111

1) Initialization<nil, W, ¢>
2) Termination<S, nil, A>
3) Left-Arc<t|S, n|I, A>→<S, n |I, A∪ {(n, t)}>
4) Right-Arc<t|S, n|I, A>→<n|t|S, I, A∪ {(t, n)}>
5) Reduce <t|S, I, A>→<S, I, A>
6) Shift <S, n|I, A>→<n|S, I, A>
7) Preposition-Shift’ <t|S, n |I, A>→<n|t|S, I, A>

The core of this algorithm is five analysis of action: LA 、RA 、RE 、SH 、PS’ . A new operation of PS’ is defined for the prepositional phrase and that is also different from the BPP algorithm. When the current top element of the stack S is a preposition and if the element that after the top element n does not match any of the following conditions: ‘de’, punctuation, adverbs and prepositions, push the element n into the stack S; if the element that after the top element n does match anyone of the following conditions: ‘de’, punctuation, adverbs and prepositions, the top element n is just the right boundary of the prepositional phrase. If the element that after the top element n is a verb, push the top element n into a new queue T instead of the stack S. Then continue to analyze the new top element of the stack I, if the element that after the top element of the stack I match any of the following conditions: ‘de’, punctuation, adverbs, verb and prepositions, the top element n of is the right boundary of the prepositional phrase; if not, push the top element of the stack I into the queue T and Until the end of a sentence can’t find an element of the stack I match the conditions, the top of the queue T is just the right boundary of the prepositional phrase.

B. *Prepositional Phrase Recognition and Treebank*

According to the following word’s features of the prepositional phrases we use SVM as a tool to be the identifier of a prepositional phrase. The following describes the features of constructing the identification of the prepositional phrase.

This paper combines Isozaki’s characteristics[12] with Chinese characteristics, after plenty of experiments, choose the 11 combinations of features for automatic identification of the prepositional phrase. The features are shown in Table I.

TABLE I: THE FEATURES

Feature	Characterization	Values
Wn	Different position of the words	1, 2---21000
Posn	The pos of different position	1---28
P_is_found	Is found the preposition (Wp)	0, 1
Distance	The distance between Wp and Wn	1—40
Wi	The word between Wp and Wn	1, 2---21000
Posi	The pos between Wp and Wn	1---28
Num_verb	The number of verb word between Wp and Wn	1—20
Num_con	The number of conjunction word between Wp and Wn	1—20
Num_pre	The number of preposition word between Wp and Wn	1—20
Num_com	The number of punctuation word between Wp and Wn	1—20

In this paper we use corpus which was changed by the Harbin Institute of Technology Information Retrieval Laboratory’s Chinese dependency-based Treebank.

C. *Algorithm Tool*

The implementation tool in this paper is Dependency parser MaltParser [13], which has the feature as Data driven, trainable, uncertainty. In order to predict the activities of the parser and select the appropriate parsing, it uses a studying strategy based on memory-and-SVM. MaltParser only needs a limited amount of training data; it can be easy and convenient to achieve a suitable new language parser. MaltParser includes the following three parts: parser, guidance and learning device. Its mission is to build a dependency graph deterministic parsing algorithm and predict the next history based on characteristics of model then mapping the history record as discrimination machine learning methods of the parser activities.

This paper adopts parsing algorithms and feature model which is different from MaltParser. The syntactic parser and the guide part of the source code have been modified, and automatic identification right boundary of the prepositional phrase which contains verbs has been realized by using this Syntactic dependency analyzer MaltParser. The algorithm parsing model was shown in Fig.1.

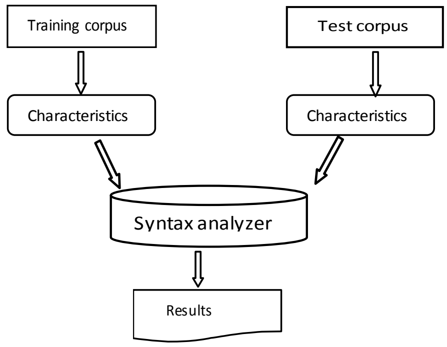

Fig. 1. Analysis model diagram

The syntactic analysis model diagram illustrates the entire process of parsing. Firstly, get the training corpus and test corpus ready to meet the requirement and then select the appropriate characteristics experiment the training data by using the syntax analyzer and a training model to test the test corpus, the syntax analyzer is a tool that can achieve the syntactic analysis algorithm.

IV. ANALYSIS OF EXPERIMENT RESULTS

A. *Experiment Results*

This paper takes the accuracy of the words that determine the parent node correctly as the evaluation of the system.

Whether dependencies are marked or not we use LA and UA to express. Test the Harbin Institute of Technology Treebank by RPP algorithm, Arc-eager algorithm and this algorithm .The test results are shown in Table.II.

TABLE II: THE FEATURES

Algorithm	UA (%)	LA (%)
Arc-eager	80.21	78.02
Bpp	81.83	79.87
This algorithm	82.91	80.76

The experiment observation indicates that using the algorithm of this paper, parsing accuracy has been improved, and the algorithm which we have proposed is effective to long-distance dependency analysis of the prepositional phrases which contains verb.

B. The Impact of the Training Corpus

In the experiment, in order to observe the impact of the corpus's size on syntactic analysis's results we gradually increase the training set, comparison of the different numbers of training set of sentences to the experimental results. Along with increasing scale of the training data, the test results are shown in Table III:

TABLE III: THE RESULTES OF DIFFERRENT CORPOR'S SIZE

Number of data set	UA (%)	LA (%)
1325	78.02	74.73
2545	80.22	76.08
4361	81.02	78.68
5537	81.62	79.59
7208	81.76	79.94

The test result indicates that: increasing the scale of training data can be useful for improving system performance. Meanwhile, with the corpus size increases, the system performance becomes increasingly slow.

V. CONCLUSION

In this paper we presented a new algorithm to solve the problem of identifying the right boundary of the prepositional phrase which contains a verb. Overall, our experiments indicate that our algorithm yields much better results in Chinese syntactic analysis than the Arc-eager algorithm and BPP algorithm. Its shortcomings indicate that future work lies in improving the executing efficiency.

ACKNOWLEDGMENT

Y.Y.Wang gratefully acknowledges professor J.L .Xu who gave her great help during the writing paper. She also thanks her friends and her family members who gave her support.

REFERENCES

[1] J. W. Gan and D. G. Huang, "Automatic Identification of Chinese Prepositional Phrase," *Journal of Chinese Information Processing*, vol. 19, pp.17-23, 2005.

[2] E. Brill and P. Resnik, "A rule based approach to prepositional phrase attachment disambiguation," in *Proc. 15th International Conference on Computational Linguistics,* Kyoto , Japan , 1994, pp.1198-1204.

[3] A. S. Yeh and M. B. Villain, "Some Properties of Preposition and Subordinate Conjunction Attachments," in Proc. *17th international conference on Computational linguistics*, Montreal, Canada, 1998, pp.1436-1442.

[4] F. Liu, T. J. Zhao, H. Yu, M. Y. Yang, and G. l. Fang, "Statics-Based Chinese Chunk Parsing," *Journal of Chinese Information Processing*, vol.14, pp. 28-32, 2000.

[5] H. Li, Y. M. Tan, J. B. Zhu, and T. S. Yao, "Chinese Chunk Recognition," *Journal of Northeastern University(Natural Science)*, vol.25,pp.114-117,2004.

[6] Y. Q .Zhang and Q. Zhou, "Automatic Identification of Chinese Base Phrase," *Journal of Chinese* Information *Processing*, vol. 16, pp. 1-8, 2002.

[7] Y. Q .Zhou, Y. K. Guo, X. J. Huang, and L.D. Wu, "Chinese and English Base NP Recognition Based on a Maximum Entropy Model," *Journal of Computer Research And Development*, vol. 40, pp. 440-446, 2003

[8] S. J. Li, Q. Liu, and Z. F .Yang, "Chunk Parsing with Maximum Entropy Principle," *Chinese Journal of* Computers, vol. 26, pp. 1722-1727, 2003.

[9] H. Li, J. B. Zhu, and T.S .Yao, "SVN Based Chinese Text Chunking," *Journal of Chinese Information Processing*, vol. 18, pp. 1-7, 2004.

[10] W. L. Yao, L. Wang, and L L. Gao, "An Ungreedy Chinese Deterministic Dependency Parsing Considering Long Distance Dependency," in *Proc. Natural Language Processing and Knowledge Engineering*, Beijing, 2008, pp. 276-280.

[11] J. Nivre, "An efficient algorithm for projective dependency parsing," in *Proc. 8th International Workshop on* Parsing *Technologies*, 2003, pp. 49-160.

[12] H. Isozaki, H. Kazawa, and T. Hirao, "A Deterministic Word Dependency Analyzer Enhanced With Preference Learning," in *Proc. 20th international conference on Computational Linguistics*, Geneva, Switzerland,2004, pp. 275-281.

[13] J. Nivre, J. Hall, J. Nilsson, and A. Chanev, "MaltParser: A language-independent system for data-driven dependency parsing," *Natural Language Engineering*, vol.13, pp. 95-135, 2007.