Asian Journal of Information Technology 5(8): 901-905, 2006
Medwell Online, 2006

A Divide-Conquer Strategy for Chinese Text Chunking

[1,2]Ying-Hong Liang, [1]Ni-Hong Wang, [1]Zhao-Wen Qiu and [1]Hong-E Ren
[1]School of Information and Computer Engineering in North East Forestry University, Harbin, 150040
[2]MOE-MS Key Laboratory of Natural Language Processing and Speech in Harbin
Institute of Technology, Harbin, 150001

Abstract: Traditional Chinese text chunking approach is to identify phrases using only one model and same features. It has been shown that the limitations of using only one model are that: the use of the same types of features is not suitable for all phrases and data sparseness may also result. In this study, the divide-conquer approach is proposed and applied in the identification of Chinese phrases. This strategy divides the task of chunking into several sub-tasks according to sensitive features of each phrase and identifies different phrases in parallel. Then, a two-stage decreasing conflict strategy is used to synthesize each sub-task's answer. Through testing on Chinese Penn Treebank, F score of Chinese chunking using Multi-agent strategy achieves to 95.23%, which is higher than the best result that has been reported.

Key words: Text chunking, sensitive features, divide-conquer strategy

INTRODUCTION

Text chunking is the key content of shallow parsing. It can be used to broad Natural Language Processing (NLP) fields: machine translation, information extraction, topic content analyzing and text processing etc. And the result of text chunking has fatal effect on the correctness of text analyzing and text processing.

Text chunking has received much attention since Abney (1991) proposed the strategy of shallow parsing[1] and designed a shallow parser[2]. The theme of CONLL 2000 was English text chunking[3]. In this conference, many statistical methods or machine learning methods were used into English text chunking[4-7]. Since 1990s, many Chinese researchers have studied on Chinese text chunking. Jun Zhao (1999) identified Chinese noun phrase using the method of transfer-based and error-driven[8]; Zhang Yiqi (2003) identified nine Chinese phrases using Memory-Based Learning (MBL) strategy[9]; Heng Li (2004) used SVM to identify Chinese phrases[10]. Li Sujian and Qun Liu (2003) gave the definition of twelve phrases and established Chinese chunking corpus from Chinese PennTree Bank[11].

To sum up above approaches, we can find that English text chunking has made a great progress to certain extent. However, Chinese text chunking is on the primary stage and there is no public corpus to researchers. But, as a strategy of shallow parsing, text chunking is the pre-processing step before attempting web information extraction, full parsing etc. so it is required higher precision, recall and faster speed. However, current result can't satisfy such requirement.

The main task of text chunking is to identify phrases in a sentence[12]. From previous study of Chinese text chunking, we know that only one model and same features were used to identify all types of phrase in these methods. Its disadvantages are as follows:

- Each phrase's characteristics couldn't be comprised by one model, so the result of some phrases (such as List phrase (LST), Quantity Phrase (QP)) are not satisfactory;
- The same type of features was used by all phrases. But some of them were not suitable to other phrases;
- Generally, A better result can be achieved if more features are used, but it will lead to data sparseness if word type of feature is used to all types of phrase.

In fact, different phrases have different sensitive features. For example, noun phrase and verb phrase are sensitive to the feature of Part of Speech (POS). However, LST constitutes of some fixed words such as —, —. So, LST is more sensitive to the feature of "word" than that of POS. Moreover, the quantity of word that constitutes of LST is limited, so data sparseness can be avoided if the feature of word is only used to those phrases that are sensitive to the feature of word.

Corresponding Author: Ying-Hong Liang, School of Information and Computer Engineering in North East Forestry University, Harbin, 150040

From above analysis, we can conclude that the result will not be satisfactory if only the feature of POS is used to identify all types of phrase in order to avoid data sparseness. Therefore, it will have good result if we select different feature and different model for different phrase so as to get the last result by integrating the identification of each phrase. Therefore, a divide-conquer strategy is proposed by us to identify Chinese phrases. This strategy divides phrases according their sensitive features. This strategy remedies the shortcomings in using only one model to identify multiple types of phrases and also has several advantages:

- It applies the theory of divide-conquer into the field of NLP and gives attention to the characteristics of each phrase, which can't be true using only one model to identify many types of phrases;
- Different models and sensitive features are used to identify different phrase, which not only avoids data sparseness but also improves the speed and performance of chunking.

CHINESE TEXT CHUNKING BASED ON DIVIDE-CONQUER STRATEGY

The definition of chinese phrases: We identify seven phrases and the simple definition is as following:

- Noun Phrase (NP): The noun phrase is headed by a noun and it optionally takes modifiers. Under our current specification, it never takes complements of any kind.
- Verb Phrase (VP): The verb phrase is headed by a verb and it optionally takes modifiers. Under our current specification, it takes complements such as , 7, .
- Adjective Phrase (ADJP): The adjective phrase is headed by adjective or the word with POS=AD (except for those included by NP).
- Adverb phrase (ADVP): The adverb phrase is headed by adverb or the word with POS=VA. (except for those included by VP)
- List Phrase (LST): Characters, letters and numbers which identify items in list, are labeled LST;
- Unnumbered lists such as dashes have to be determined by the context and they may occur either within one sentence or multiple sentences. When the list items, enumerated or not, occur in separate sentences, (as indicated by a period or some other kind of punctuations), treat the colon as the final punctuation and place each list item in its own set of empty outer parenthesis.
- Quantitative Phrase (QP): The quantitative phrase is headed by numeral and its quantifier;
- Preposition Phrase (PP): The preposition phrase is headed by preposition or some words represent direction or position.

The model of chinese text chunking based on divide-conquer strategy: It is the main content of divide-conquer theory to solve a problem in a decomposing way. Task share and result share are often used to get this answer. The procedure of getting the answer of a question is divided into three parts:

- Decomposing the task: The task required to be decomposed into sub-tasks;
- Getting the answer of sub-task£°To get each sub-task's answer separately;
- Synthesizing each sub-task's answer: Integrating sub-task's answers.

In this study, a divide-conquer strategy was proposed to identify Chinese phrases and consider sensitive features of each phrase. The model of Chinese text chunking using divide-conquer strategy is shown in Fig. 1.

Above partition is not exclusive. From Figure 1, we can find that chunking task is divided into 4 sub-tasks. Then each sub-task's answer is to be achieved separately. The right part in Figure 1 is to synthesize each sub-task's answer.

The implementation of divide-conquer strategy for chinese text chunking: In our system, seven types of phrases are identified. Table 1 briefly outlines each task.

The sensitive features of each phrase: A sensitive feature is the feature that has a crucial effect for chunking. In the large quantity of features, there are only several features that are crucial for chunking and other features are useless. These redundant features not only take up the memory space but also affect the searching efficiency. So, larger quantities of features not always get better result. The most important is whether the feature is a sensitive feature. We analyze the constituent of each phrase in training corpus and summarize sensitive features of each phrase as Table 2:

In Table 1: W =current word; P_{left1}= previous POS; =P Current POS; P_{right1} = next POS; P_{string}= the POS string that can compose a phrase; C_{left1}= the previous chunk type; C_{right1}= next chunk type; $PRO_{boundary}$ = boundary probability.

The two-stage strategy for decreasing the conflicts among phrases: In the course of synthesizing each sub-task's answer, conflicts will occur among some phrases. Some conflicts are listed as follows:

- QP and LST

These two phrases all use the feature of "word". But, some words perhaps are part of QP phrases and they are perhaps are part of LST phrases too. So, there are conflicts between QP and LST. For example:
Sentence 1: QP(CD 三) (M 年)ADVP (AD 多)O(LC 来) (PU,)……

Sentence 2: LST(CD 一)VP(V 是)NP(NT 一九九六年)(NR ャキ)VP(AD オホカ)(VV モネ)NP(NN オネ)O(PU ,) ……

- ADVP and VP

In a general way, we regard a word as part of ADVP phrase if its part of speech is "AD". But, some words are part of VP phrases although their part of speech is "AD". So, there are conflicts between ADVP and VP. For example:

Sentence 1: ADVP (AD ネ)VP(AD シ) (VV ヤ)NP(NN キ ス) (NN サホカ) O (PU ,) ……

Sentence 2: ADVP (AD モ)VP(AD モ)(VV サシ) ……

Other phrases, such as ADJP and NP, QP and NP, can also occur conflicts. Decreasing the conflicts among phrases will benefit to improve the recall and precision of Chinese chunking. Thus, we proposed the two-stage decreasing conflicts strategy. The procedure of two-stage decreasing conflicts strategy is as follows:

- First, decreasing conflicts using rules that summarized manually in advance; for example:

Rule 1 for decreasing conflicts between QP and LST: QP is followed by fixed quantifier, while LST is followed by punctuation.

Rule 2 for decreasing conflicts between ADVP and VP: a word with "AD" part of speech usually ties to some fixed verb.

- Then, decreasing conflicts by priority.

Through the stage 1, some conflicts have been removed, but there must be some conflicts have been leaved out. In this stage, priority is used to solve this problem. We believe that the result of higher priority is more credible than that of the lower priority. The priority of phrases is set as follows: PP> LST =QP> NP=VP> ADJP=ADVP.

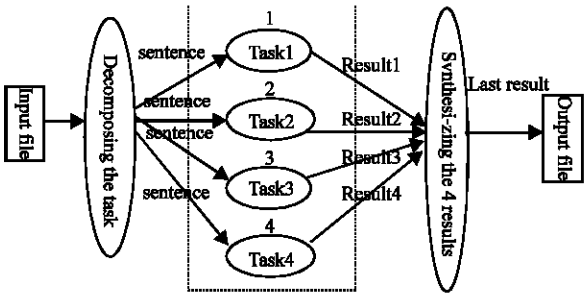

Fig. 1: The model of Chinese text chunking using divide-conquer strategy

Table 1: A brief introduction to each task

Sub-tasks	Algorithm	Function
Task 1	combination of boundary statistics and rule revise[13]	To identify NP
Task 2	longest string matching	To identify VP
Task 3	binary search and longest string matching	To identify PP, ADVP and ADJP
Task 4	longest string matching	To identify QP and LST

In Chinese, there is almost no conflict between PP and other phrase. Therefore, we set the highest priority for PP. The features of LST and QP are "word", which often are fixed word and the amount is limited. So serious data sparseness does not appear and the good result is gotten. LST and QP are easily regarded as noun phrase, so its priority was set higher than that of noun phrase. The frequency of NP and VP in a sentence is high, so it is important to identify NP and VP. ADJP and ADVP are difficult to identify because the word with POS=JJ is often in noun phrase, while the word with POS=AD is often in verb phrase. So their priority is the lowest.

The experimental result and analysis: Our corpus is Chinese PennTree Bank (chtb001- 216 is training corpus and chtb217-325 is test corpus). Chinese seven phrases are identified and Table 2 is the result comparison between divide-conquer strategy and MBL method[10]. Although these two methods used different corpus, but the definition of Chinese phrases in these two methods is similar in some extent. So, we compare the result of these methods.

In Table 3, the left Phrase type is equivalent or probably equivalent to the right one. Table 2 shows that most of the F scores using divide-conquer strategy are higher than those of MBL method. This illuminates that divide-conquer strategy can improve the result of all phrases. It is worthy to notice that the improvement of LST is larger than others. Its reason is that the feature of "word" was used by LST phrases in divide-conquer strategy and the quantity of these words is limited, so data sparseness does not occur. Another large improve is

Table 2: The sensitive features of each phrase

	Sensitive features							
		POS				Chunk type		
Phrase	W	P_{left1}	P	P_{right1}	P_{string}	C_{left1}	C_{right1}	$C_{boundary}$
NP					√		√	
VP					√			
PP	√	√						
ADVP, ADJP			√			√	√	
QP , LST	√	√		√				

Table 3: The result comparison of MBL and divide-conquer strategy

MBL				divide-conquer strategy			
Phrase type	Precision (%)	Recall (%)	$F_{\beta=1}$ (%)	Phrase type	Precision (%)	Recall (%)	$F_{\beta=1}$ (%)
Ap	92.10	88.50	90.30	ADJP	98.32	82.69	89.83
Dp	97.10	98.00	97.50	ADVP	97.75	97.86	97.80
Mbar	72.00	53.20	61.10	LST	100.00	100.00	100.00
Np	95.00	93.00	94.00	NP	95.52	98.76	97.11
Sp	74.10	70.10	72.10	PP	96.87	90.56	93.61
Vp	96.70	96.60	96.60	VP	95.65	97.87	96.75
Mp	94.50	92.30	93.40	QP	94.58	97.67	96.82
all	95.20	93.80	94.50	all	94.68	95.78	95.23

PP phrase, the reason is that there is almost no conflict between PP phrase and others. These prove that using different feature for different phrase can avoid data sparseness in divide-conquer strategy. In addition, the results of other phrases are improved in a certain extent, which is owned to adopt priority. Moreover, divide-conquer strategy has following advantages over MBL method: divide-conquer strategy only uses sensitive features, so the quantity and type of features are few and occupy little memory. The time cost of the proposed method is also bearable. We tested the speed of this method in the computer whose CPU is PIV 2.5G, with 256M memory. The speed is 865w/s (865 words per second).

CONCLUSION

Chinese chunking is the core part of shallow parsing. Researchers have previously paid more attention to the various approaches for chunking. However, the characteristics of phrases and the relationship between phrases are ignored. In this study, a divide-conquer chunking strategy is proposed. This strategy uses different model and sensitive feature for different phrase, whose characteristics of each phrase are sufficiently considered. At the same time, data sparseness is avoided with sensitive features. The result of chunking is improved through co-reference of phrases. The performance and speed are improved obviously on comparing with other approaches. The proposed method offers a brand-new strategy for chunking and gives good result.

The present study mainly proposes and implements a chunking model with divide-conquer strategy and there is much room to improve. Our future study is to search new algorithms that suit to identify each phrase (especially noun phrase and verb phrase) to further increase the performance of the model.

ACKNOWLEDGEMENT

This study was supported by the Natural Science Foundation of China (Grant No. 60373101); the Young Science Foundation of Harbin (Grant No. 2005AFQ X J 0 20).

REFERENCES

1. Abney, S., 1991. Parsing by chunks, Principle Based Parsing, Berwick, Abney and Tenny (Eds.), Kluwer A. Publishers.
2. Abney, S., 1996. Partial parsing via finite-state cascades, Studyshop on Robust Parsing, 8th European Summer School in Logic, Language and Information, conference, Prague, Czech Republic, pp: 8-15.
3. Tjong Kim Sang, E.F., 2000. Introduction to the CoNLL-2000 Shared Task: Chunking, Proceedings of CoNLL-2000 and LLL-2000, conference, Lisbon, Portugal, pp: 127-132.
4. Skut, W. and T. Brants, 1998. A maximum-entropy partial parser for unrestricted text, In Proceedings of the 6th Workshop on Very Large Corpora, conference, Montreal, Quebec.

5. Kudoh, T. and Y. Matsumoto, 2000. Use of Support Vector Learning for Chunk Identification, Proceedings of CoNLL-2000 and LLL-2000, conference, Lisbon, Portugal, pp: 127-132.
6. Tjong Kim Sang, E.F., 2000. Memory-Based Shallow Parsing, In proceedings of CoNLL-2000 and LLL-2000, conference, Lisbon, Portugal, pp: 559-594.
7. Zhang, T., F. Damerau and D. Johnson, 2002. Text Chunking based on a Generalization of Winnow, Machine Learning Res., 2: 615-637.
8. Jun Zhao and ChangNing Huang, 1999. The model of Chinese base noun phrase identification based transfer, J. Chinese Inform. Processing, pp: 13.
9. YiQi Zhang and Qiang Zhou, 2003. The auto identification of Chinese base phrase, J. Chinese Inform. Processing, pp: 16.
10. Heng Li, JingBo Zhu and TianShun Yao, 2004. The Chinese chunking using SVM, J. Chinese Inform. Processing, 18: 1-7.
11. SuJian Li and Qun Liu, 2003. The definition and establish of Chinese phrases, JSCL-2003, conference, Harbin, pp: 100-115.
12. Hong Lin Sun and ShiWen Yu, 2000. The summarization of shallow parsing method, the linguistics of the Present Age, pp: 063-073.
13. Ying Hong Liang and TieJun Zhao, 2004. The Identification of English Base Noun Phrase Based on the Hybrid Strategy, Computer Engineering and Application, 40: 1-8.