Journal of Convergence Information Technology
Volume 5, Number 10. December 2010

Chinese Chunking and Consistency Checking Using Rule-Based Method

Lu Jiao-Li[*1], Zheng Jia-Heng[*2], Tan Hong-Ye[2], Sun Jian[3]

[*1] *Corresponding author* *Center of Modern Education & Technology, Shanxi University, Taiyuan, China; School of Computer Science & Technology, Shanxi University, Taiyuan, China*

[*2] *School of Computer Science & Technology, Shanxi University, Taiyuan, China*

3. Beijing Ali Group R & D Center, Beijing, CHINA

lujl@sxu.edu.cn, jhzheng@sxu.edu.cn

doi:10.4156/jcit.vol5. issue10.2

Abstract

This paper presents a rule-based chunking approach. Rule-based method does well in analyzing the structure of natural language. In order to avoid the confliction of the rules, we extract a small scale chunking rule set for chunking first. Then we define more rules to check and correct the inconsistency phenomena. We also adopt man-machine interaction method to solve some special language phenomena. Experimental results show that our approach achieves high accuracy.

Keywords: *Chinese Chunking, Consistency Checking, Rule-Based Approach*

1. Introduction

With the advent of the Internet era, many technologies of language processing are widely used to process the amount of electronic data from web. Among these technologies, parsing is basic, necessary and plays an important role in natural language processing. However, a full parser is usually costly and slow and the performance is not suitable for online applications [2]. As an alternative to full parsing, shallow parsing is much more efficient and less demanding, and currently, it has been applied to various natural language processing systems [3], such as information retrieval, information extraction, question answering, and automatic document summarization, because shallow parser is more suitable for theses realistic applications.

Shallow parsing, also named partial parsing or chunking, is recognized as series of processes. First, identifying proper chunks from a sequence of tokens (such as words), and second, classifying these chunks into some grammatical classes. A lot of techniques have been developed for shallow parsing. Most of these techniques base on statistically approaches such as SVM, CRF, and ME [4, 5, 6], as for English texts, the parsing speed of each approach is fast and the parsing accuracy is acceptable, but as for Chinese texts, the accuracy is still not very high. That is because Chinese is very flexible and intractable, and statistically based algorithms always cannot care about it.

Section 2 describes our shallow parsing standard in detail. Section 3 presents the rule-based approach. Section 4 consistency checking method is introduced. Section 5 presents some experiment results of chunking. Finally, we draw some conclusions.

2. Shallow Parsing Standard

Currently, there is no standard for Chinese shallow parsing. There are also some works on a standard for Chinese shallow parsing [7, 8], nevertheless, the POS standard and vocabulary in each approach are different. Here we make our standard by referring to the chunking criteria of Tsinghua University and State Language Commission. We use the following ten tags for chunking.

Table 1. Chunk Tags

Chunk Tag	Example
NP	[救市/vn 资金/n]NP
VP	[努力/ad 学习/v]VP
AP	[高兴/a 得/ue 很/dc]AP
MP	[一百/m 多/m 万/m]MP
QP	[1 0 万/m 多/m 斤/qd]QP
FP	[大型/f]FP 和/c [中型/f]FP
DP	[不/df 太/dc]DP
PP	[[从/p]PP [今天/t]TP]YPP
TP	[1783 年/t 2 月/t]TP
SP	[[中国/ns]NP [内地/s]SP]YSP

In this paper, we regard each word as a token. Since a phrase can have more than one token, we associate two tags, x: x_begin and x_continue, with each category. In addition, we use the tag "NONE" to indicate that a token is not part of a phrase. The shallow parsing problem then can be redefined as a problem of assigning one of the 2n+1 tags to each token. This is called the B-I-O scheme. There are ten named entity categories and 21 tags: NP_begin, NP_continue, VP_begin, VP_continue, PP_begin, PP_continue, AP_begin, AP_continue, MP_begin, MP_continue, QP_begin, QP_continue, DP_begin, DP_continue, TP_begin, TP_continue, SP_begin, SP_continue, FP_begin, FP_continue, and NONE. Finally, we transform the 21 tags into "[" and "]", thus, we can view chunking results conveniently.

3. A rule-based Shallow Parser

Since we have no large scale data which is chunked by chunk tags, and at the same time, statistically based methods have some innate weakness to analyze the structure of natural language, here we apply a rule-based method to process the data. Firstly, we randomly select 500 sentences and chunk them manually. Then we sum up the rules by analyzing the chunking results of these 500 sentences. In turn, we apply these rules to the rest data and guide machine to chunk them.

3.1. Construction of Chunking Rules

There are two methods to obtain chunking rules. One is manually constructed by specialists, and the other is extracted from the chunked corpus. We adopt the latter. Here we give three formal definitions for chunking rules.

Definition 1, Basic Chunking Rule, if exists a sequence w1/t1, w2/t2...wm/tm to construct a chunk in a sentence, the part-of-speech tag sequence t1, t2...tm is called a Basic Chunking Rule.

Definition 2, Conjunctive Basic Chunking Rule, firstly, we set a threshold θ. If the frequency of a Basic Chunking Rule, f(r), meet the inequality f(r)> θ, we name the Basic Chunking Rule a Conjunctive Basic Chunking Rule.

Definition 3, Basic Chunking Rule Set G, a Basic Chunking Rule Set G is made up of all the Conjunctive Basic Chunking Rules.

It's easy to get Basic Chunking Rules from chunked sentences. Scan the chunked sentences from left to right, and we can extract a Basic Chunking Rule from every chunk structure. At the same time, we statistic the frequency of these Basic Chunking Rules to decide whether they are Conjunctive Basic Chunking Rules and whether they are components of Basic Chunking Rule Set G. Basic Chunking Rules are composed of part-of-speech tags. Combine with the definition of the phrase, chunking rules have the following form.

1) NP→a n for example, [好/a 姑娘/n]NP
2) NP→m * q n for example, [两/m 个/qe 解/n]NP
3) NP→b n for example, [女/b 将领/n]NP

4) VP→[d *] v for example, [频频/d 点头/v]VP

5) VP→d 地 v for example, [默默/d 地/ui 观察/v]VP

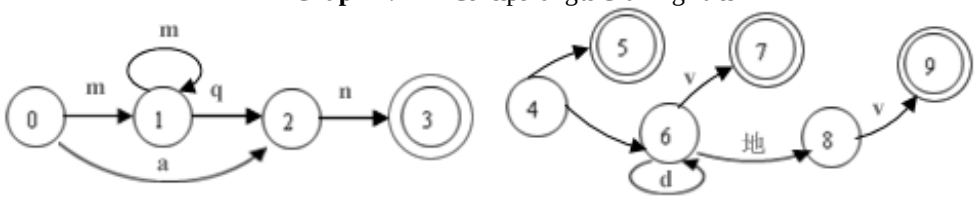
*stand for one or more, and [ ] stand for it can be ignored.

3.2. Chunking Based on Finite-State Cascades Method

Abney put forward finite-state cascades method for chunking. Based on this principle, we construct the Deterministic Finite Automaton (DFA) to analyze the input sentences. A DFA corresponds to a Conjunctive Basic Chunking Rule, and a Conjunctive Basic Chunking Rule can be looked as a pattern, so the process of chunking is in fact a process of pattern matching.

Corresponding DFA to the above five chunking rules are demonstrated in the following graphs.

Graph 1. DFA Corresponding to Chunking Rules



◯stand for a state of DFA, ◎stand for the end of DFA.

We'd like to describe the specific process. First, we input the Chinese sentences which are segmented and part-of-speech tagged. Then we try to search out the best-matched language block from an initial state of the DFA. When DFA can reach terminal state, a chunk will be generated. On the other hand, if we meet a word that can not generate a chunk, there is no influence, we simply skip this word and continue our process.

4. Consistency Checking

The significant advantage of rule-based methods lies in that they make full use of the language knowledge, but small scale rule set may lead to rough and not very fine results. Some negative effects, especially inconsistence are generated. So, it's necessary to add more rules and check the consistence and correct the chunking results. Our inconsistency checking includes consistency checking for segmentation, for part-of-speech tagging, and for chunking.

4.1. Consistency Checking for Segmentation

Now, segmentation software can achieve a high accuracy. However, inconsistency phenomena still exist. Sometimes, the same string, in the same context, often has the different segmentation results. For example, "有点/d 难/a" and "有/vx 点/qb 差/a", the words "有点" in the same contextual environment, but the segmentation result is different. In order to solve this problem, we propose the following steps. First, extract base words from corpus. Second, obtain combinational words and compare them with base words, if search out the same words from base vocabulary, we regard these words as the inconsistently segmented words. Then we save these inconsistency words and sentences. Finally, correct the segmentation results by orderly judging the context of inconsistency words.

In addition, we also add more rules for the segmentation of some special conditions. For example, "v+不了", "v+得到", "v+不住", "v+不到", "v+不+v", we combine these Three-character Words into a word. That is, like the Three-character Words "经不住", "动不动", "受不了", we look them as a segmentation unit.

4.2. Consistency Checking for Part-of-speech Tagging

The inconsistence of part-of-speech tagging mainly lies in multiple-category words. Here, we only think about the circumstance that the same word, which in the same context, but the result of the part-of-speech tagging is different. For example, "贺敬之/nr 下笔/vi 之/rz 时/Ng" and "栖身/vi 之/u 所/n", the phrase has the same structure, but the word "之" have the different part-of-speech tag. Our tactics to this problem associate the following steps.

1) Extract all the multiple-category words
2) Obtain the sentences with the same multiple-category word
3) Use vector to express the POS sequence of these sentences
4) Compute the similarity of vectors to classify them, the multiple-category words with the same class should have the same POS tags.
5) Obtain the center vector of every class to decide the POS tags of the multiple-category words

Similar to segmentation, we still add some rules for the POS tagging of some special conditions. One of the rules we added is that the words next to "的" should be a nominalization words. For example, "对/p 形象思维/n 的/ud 研究/vn", "我/rr 喜欢/v 她/rz 的/ud 美丽/an", the part-of-speech tag of the word "研究" and "美丽" should be transformed from "v" and "a" to "vn" and "an".

4.3. Consistency Checking for Chunking

We check the inconsistence of chunking mainly by means of adding more rules. Compared with the rules used by shallow parser, here associate more difficult language phenomena to ensure a more fine chunking result. Here are five examples of our rules.

1) "v +一+ v", directly construct a verb chunk, for example, [谈/v 一/m 谈/v]VP
2) "是+a/v+的", "a/v+的" should construct a noun chunk, for example, [是/vl]VP [[好/a]AP 的/ud]YNP
3) "r+a", "r" and "a" construct a noun chunk and a adjective chunk respectively, for example, [这么/rz]NP [大/a]AP
4) "m+q+m", directly construct a quantifier chunk, for example, [四/m 尺/qd 多/m]QP [深/a]AP
5) "n+m", construct a noun chunk and a numeral chunk respectively, then combine them into a noun chunk, for example, [见/v]VP [[图/n]NP [2/m]MP]YNP

4.4. Man-machine Interaction

We adopt man-machine interaction method to process some intractable problems, such as the identification of the newly generated words, parallel relation of the chunks, the problem of long distance dependency, VN and VO constructions, etc.

5. Experiment

We use ALIBABA Treebank as our chunking corpus. It contains 40,000 sentences, obtained from People's Daily, B2B corpus of ALIBABA, etc. All the sentences share the same standard and do the same consistency checking. The statistical data of inconsistence is demonstrated in the following table.

Table 2. The Amount of Inconsistency sentences

	The amount of inconsistency sentences
segmentation	1680
POS tagging	5834
chunking	2547

We corrected all the inconsistency points, until we could not search out them. In addition, we obtained fine results for some intractable problems. Here we list some shallow parsing results for our model in the following table.

Table 3. Some of Our Chunking Results

Some Results of Chunking
盆地/n 中部/f 有/vx 一/m 条/qe 东西向/jn 红色/n 砂岩/n 构成/v 的/ud 低/a 山/n ***[[盆地/n]NP [中部/f]SP]YSP [有/vx]VP [一/m 条/qe]QP [东西向/jn 红色/n 砂岩/n]NP [构成/v]VP 的/di [低/a 山/n]NP***
二氧化碳/n 是/vl 一/m 种/qe 密度/n 比/p 空气/n 大/a 的/ud 无色/n 气体/n ***[二氧化碳/n]NP [是/vl]VP [一/m 种/qe]QP [密度/n]NP [[比/p]PP [空气/n]NP]YPP [大/a]AP 的/ud [无色/n 气体/n]NP***
词义/n 总/d 是/vl 指称/v 某/r 类/qz 事物/n 和/c 现象/n 的/ud ***[词义/n]NP [总/d 是/vl]VP [指称/v]VP [某/r 类/qz]QP [[事物/n]NP 和/c [现象/n]NP]YNP 的/ud***
一千/m 年/qt 来/f，/wd 人类/n 历史/n 发生/v 了/ul 沧桑/n 巨变/vn ***[[一千/m 年/qt]QP [来/f]TP]YTP ，/wd [人类/n 历史/n]NP [[发生/v]VP 了/ul [沧桑/n 巨变/vn]NP]YVP***
我方/rr 冒昧/ad 致函/v，/wd 请/v 贵/a 公司/n 惠/Vg 送/v 贵/a 方/Ng 在/p 广告/n 中/f 所/us 推销/v 产品/n 的/ud 样品/n ***[我方/rr]NP [冒昧/ad 致函/v]VP ，/wd [[请/v]VP [贵/a 公司/n]NP]YVP [惠/Vg 送/v]VP [贵/a 方/Ng]NP [在/p 广告/n 中/f]PP [所/us [推销/v]VP [产品/n]NP]YNP 的/ud [样品/n]NP***

6. Conclusion

In this paper, we proposed a rule-based Chinese shallow parser that can chunk Chinese sentences into ten chunk types. We also added more rules to check and correct the inconsistence of our chunking results. In addition, adopt man-machine interaction method to correct the Chinese chunking corpus mainly aimed to some difficult problems. All of these ensured the high accuracy of our results. In the future, we will continue to improve our efficiency by adopting some machine learning approaches to automatically correct the chunking results. In addition to ALIBABA Treebank, we will extend our training corpus by incorporating other corpora, such as Penn's Chinese Treebank to guarantee the robustness of our model.

7. Acknowledgement

This work is supported by National Natural Science Foundation 60775041.

8. References

[1] Abney, S.P. Parsing by Chunks. in Berwick, R.C., Abney, S.P. and Tenny, C. eds. Principle-Based Parsing: Computation and Psycholinguistics, (Kluwer, Dordrecht, 1991), 257-278.
[2] XIA, X. and WU, D., Parsing Chinese with an almost-context-free grammar. in EMNLP-96, Conference on Empirical Methods in Natural Language Processing, (Philadelphia, 1996).
[3] Müller, F.H. and Ule, T., Annotating topological fields and chunks - and revising POS tags at the same time. in Nineteenth International Conference on Computational Linguistics (COLING 2002), (Taipei, Taiwan, 2002), ACM, 695-701.
[4] F. Sha and F. Pereira. Shallow parsing with conditional random fields. Proceedings of Human Language Technology Coference'2003, (Edmonton, Canada, May 27-June 1, 2003), 134-141.
[5] T. Kudo and Y. Matsumoto, Use of Support Vector Learning for Chunk Identification, Proceeding of CoNLL-2000 and LLL-2000, (Lisbon, Portugal, 2000), 142-144.
[6] G. Sun, C. Huang, X. Wang and Z. Xu. Chinese Chunking Based on Maximum Entropy Markov Models. International Journal of Computational Linguistics and Chinese Language Processing, (2006), 115-136.

[7] Li, B., Lu, Q. and Li, Y., Building a Chinese Shallow Parsed Treebank for Collocation Extraction. in CICLing, (2003), 402-405.
[8] Xu, R.-F., Lu, Q., Li, Y. and Li, W., The Construction of A Chinese Shallow Treebank. In the Third SIGHAN Workshop on Chinese Language Processing, (2004), 94-101