Name of the workshop: Workshop on Artificial Intelligence

Title of the paper: A Way to Assign Parts-of-Speech Information to
Chinese Frequent Strings

Abstract:

A CFS is a frequently used combination of Chinese characters which have been defined in our previous research [11]. A CFS may be a proper noun, like “**網際網路**” (the Internet), a verb phrase, like “**全力動員投入**” (try one’s best to mobilize), and so on. If a CFS can have some kinds of POS (part-of-speech), we can use it in more applications. In this paper we propose a method to assign the part-of–speech information to CFSs. If a CFS *s* is also a word *w*, we can assign the POSs of *w* to *s*. When *s* is a combination of several words, we will try to find some possible POSs associated with it. We use the Sinica Treebank which contains 38,725 parsing trees as our training and testing corpus. We extract 15,946 parsing rules from 90% of the 38,725 parsing trees. There is 10% of the corpus left for outside test. The accuracies of outside test of assigning POSs to CFSs are 71.02% and 98.81% for top 1 and top 5 choices, respectively.

Authors: Yih-Jeng Lin[1], Feng-Long Huang[2], and Ming-Shing Yu[3]

[1]Department of Information Management, Chien Kuo Institute of Technology
Changhua, 500 Taiwan. yclin@amath.nchu.edu.tw

[2]Lab. of Natural Language Processing, National Lian-Ho Institute of Technology
Miao-Li, 360, Taiwan. flhuang@mail.nlhu.edu.tw

[3]Department of Applied Mathematics, National Chung-Hsing University
Taichung, 402 Taiwan. msyu@dragon.nchu.edu.tw

Contact author: Yih-Jeng Lin

Tel:+886-4-7224676#3600, Fax:+886-4-7291952, +886-4-22622310

Email: yclin@amath.nchu.edu.tw

Keywords: Chinese Frequent Strings, Part-of-speech, Treebank, and
Parsing.

A Way to Assign Parts-of-Speech Information to Chinese Frequent Strings

Yih-Jeng Lin[1], Feng-Long Huang[2], and Ming-Shing Yu[3]

[1]Department of Information Management
Chien Kuo Institute of Technology
Changhua, 500 Taiwan
[2]Lab. of Natural Language Processing
National Lian-Ho Institute of Technology
Miao-Li, 360 Taiwan
[3]Department of Applied Mathematics
National Chung-Hsing University
Taichung, 402 Taiwan

Abstract

A CFS is a frequently used combination of Chinese characters which have been defined in our previous research [11]. A CFS may be a proper noun, like “**網際網路**” (the Internet), a verb phrase, like “**全力動員投入**” (try one’s best to mobilize), and so on. If a CFS can have some kinds of POS (part-of-speech, **詞性**), we can use it in more applications. In this paper we propose a method to assign the POS information to CFSs. If a CFS *s* is also a word *w*, we can assign the POSs of *w* to *s*. When *s* is a combination of several words, we will try to find some possible POSs associated with it. We use the Sinica Treebank which contains 38,725 parsing trees as our training and testing corpus. We extract 15,946 parsing rules from 90% of the 38,725 parsing trees. There is 10% of the corpus left for outside test. The accuracies of outside test of assigning POSs to CFSs are 71.02% and 98.81% for top 1 and top 5 choices, respectively.

Keywords: Chinese Frequent Strings, Part-of-speech, Treebank, and Parsing.

1. Introduction

We have defined a CFS to be a string, which appears more than once [11]. There are some combinations of Chinese characters that are as important as unknown words. Such combinations appear frequently in Chinese texts. We call the combinations Chinese frequent strings (CFSs). And the unknown words are a fraction of the Chinese frequent strings (CFSs). We also have shown that using CFSs can be helpful in solving some problems of Chinese natural language processing (NLP), such as Chinese phoneme-to-character conversion and Chinese character-to-phoneme conversion [11].

There are some researches related to the idea of CFSs. Many researches focus on the extracting unknown words and proper nouns [1, 2, 3, 5, 17]. The POSs of most unknown words and proper nouns are nouns. Since a CFS is a string which may contains some Chinese characters and it may not be a noun, a method to assign POSs to a CFS is required. For instance, consider the Chinese segment

"**舉個例子來說** (Taking an example)"

The segment "**舉個例子來說**" is a CFS but not a unknown word, it is a frequently used string. Such CFS should have some other kinds of POSs. If we can assign possible POSs to CFSs, it may be helpful for natural language processing, such as parsing.

In this paper we use the Sinica Treebank [9] as our training and testing corpus. Sinica Treebank (**中研院中文句結構樹**) is a corpus which contains 38,725 parsing trees. Each tree of sentence contains the syntactic and semantic information, such as lexicon, POSs, semantic role and tree structure. Such information is helpful for natural language processing. We will try to extract some parsing rules from Treebank automatically. And we proposed a method to assign the POSs to a CFS by using such parsing rules.

The organization of this paper is as follows. In Section 2, we will show some related works. The corpus and dictionary we used will be introduced in Section 3. Section 4 shows the extracting of the parsing rules. We will propose our method to assign POSs to CFSs in Section 5. The experiment results will show in Section 6. And Section 7 is the conclusions and future works.

2. Some Related Researches

The work we do is similar to the parsing of a sentence. Some related researches about parsing are as follows.

The author of [10] divide the automatic learning approaches into two types: lazy learners [10, 12, 13] and eager learners [14, 15, 16]. The former keep all learned data available for reasoning and descent from the *k*-nearest neighbor (*k*-NN). The latter, however, extract knowledge structures or statistical information from the training data and then reason on the basis of these abstraction but not the original data.

Since the first appearance of Treebank, there are many contributions on such resources for parsing (in English or other languages). A standard method is to convert the sub-trees represented in the Treebank into stochastic phrase structures grammars. Eager learners of Treebank are so-called Treebank Grammar (TG). The Treebank grammars derive stochastic phrase structure rules for usage of traditional parsers. This approach tends to achieve better performance than that derived by hand-made grammars.

Charniak [7] derived a probabilistic context-free grammar (PCFG) from a 1,000,000 word hand-annotated corpus. Only the word's part-of-speech (POS), no other lexical information, is adopted in this strategy. Appropriate 16,000 rules are derived from the corpus. There are two drawbacks. The first is that no lexicons (word) is used. The second is the over-generation problem; since it is difficult that the parser assign the correct parse to sentence only by using the POS data.

The paper [8] constructed a model which used both the syntactic and semantic information to parse the input sentence while all the information of Treebank [9] is not adopted. At that time, the Treebank corpus [9] for Mandarin NLP is not released. [9] contains more information for Mandarin parsing, such as semantic structure of sentence and semantic roles of lexicon, and so on.

The authors [14] described an example-based parser for Chinese. Tree structure whose size is equal or smaller than the sentence to be parsed are retrieved from a Treebank and aligned with the sentence. Subsequent structural adaptation handle unknown words, type shifting and metaphorical extensions of words. Derivational adaptation reanalyze awkward subtrees in order to auto-correct badly matched trees and insert unmated, previously deleted words.

Currently, there are several works for Mandarin parsing [8, 14]. Author in [8] used the syntactic and semantic information to parse the Mandarin sentence without the grammar rules in Treebank. Lexical feature-based grammar formalism, called Information-based Case Grammar (ICG), is adopted for the parsing model, which used a core grammar (G_0) to cover the set of normal sentences and a method of grammar extension (G_1~G_n) to cover abnormal sentences. Authors in [14]; however, adopt the Sinica Treebank by using the example-based approach, in which the fuzzy match and adaptation are two principle procedures. The parsed sentences may contain some mistakes of lexical categories and sentence structure. These errors which caused by fuzzy match can' t be corrected during the adaptation procedure for some situation, especially when unknown words occurred in sentence.

3. The Treebank and ASCED

There are two corpora and a lexicon dictionary used in our research. The two corpora are Sinica Treebank v1.0 and Academic Sinica balanced corpus (ASBC，**中研院平衡語料庫**) [4] which are published by Academia Sinica, Taiwan. The lexicon is Academic Sinica Chinese electronic dictionary (ASCED，**中研院八萬目詞**) which is also published by Academic Sinica.

The Treebank contains 9 text files, and there are 38,725 parsing trees of Chinese sentences. Such Chinese sentences are selected from ASBC. Each parsing tree contains some important information such as POSs of each word, semantic role, and tree structure. We will use the some of the information to extract the parsing rules from the parsing tree of Treebank.

Since each parsing rule is consisted of some POSs, we should have a dictionary which can offer the POS information for each defined word. The ASCED is a well-defined lexicon which contains about 80,000 words with POSs for each word respectively.

In our experiments, we find that the POSs of some words defined in ASBC cannot be found in ASCED. We try to collect all possible POSs of each word in our dictionary. The POSs defined in ASBC for each word which cannot be found in ASCED should be collected.

Figure 1 is the overall structure of our approach and we can find the usage of Treebank, ASBC, and ASCED in our approach. We try to collect possible POSs for each defined word by training ASBC and ASCED. And we try to get the possible parsing rules from the parsing trees of Treebank.

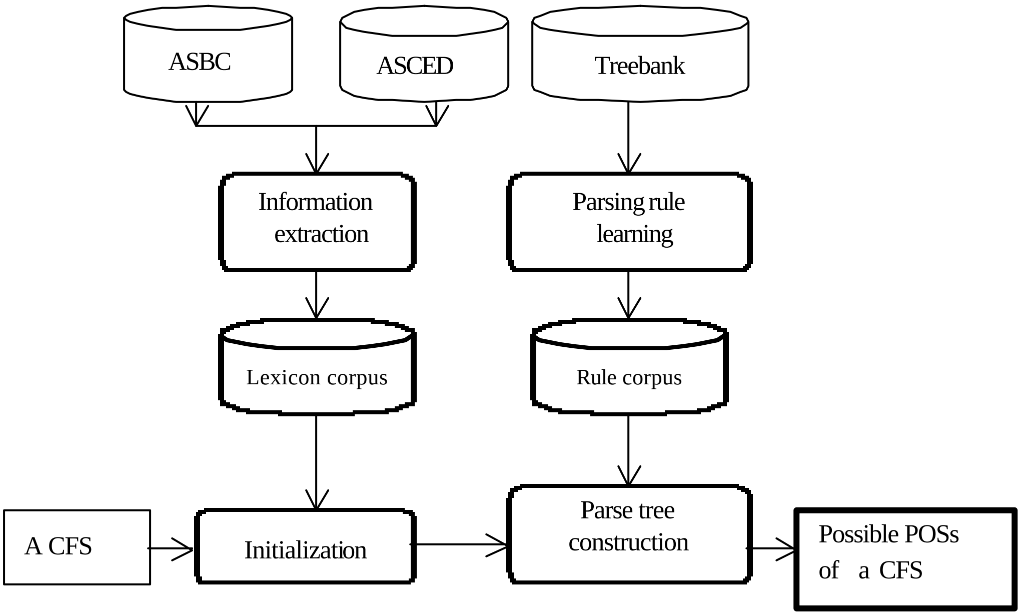

Figure 1: The overall structure of our proposed approach.

4. Extracting the Parsing Rules from Treebank

We use Sinica Treebank [6] as the training and testing data. The contents of the Sinica Treebank are the structure trees of sentences. The structure trees contain the information of words, the POS information of each word, and the reduction of the POS of some words. Figure 2 shows the structure tree of the sentence "**你要不要這幅畫**" (Do you want this picture?). The representation of this structure tree in Sinica Treebank is as follows:

#S((agent:NP(Head[1]:Nhaa:**你**))|(Head:VE2(Head:VE2:**要**)|(negation:Dc **不**)|(Head:VE2:**要**))|(goal:NP(quantifier:DM:**這幅**)|(Head:Nab:**畫**)))#

[1] In Treebank, "Head" is called "semantic role" which contain several valuable information for NLP.

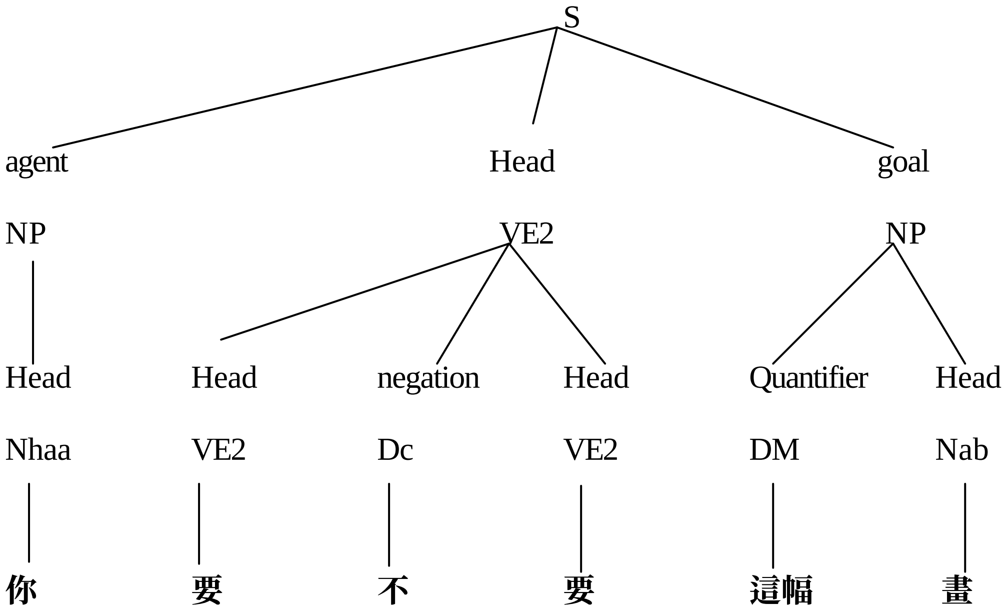

Figure 2: The structure tree of the sentence “**你要不要這幅畫**” (Do you want this picture?).

There are 38,725 structure trees in Sinica Treebank version 1.0. They are stored in 9 files. We first use a portion of the 38,725 structure trees as the training data. We want to extract the parsing rules from each structure tree. Such parsing rules are the rules for determining the POSs of CFSs. Since each CFS may contain one or more words, a POS of a CFS may be a portion of the structure trees. For example, there are 4 different parsing rules we extract from the structure tree of Figure 2. They are listed in Table 1. The notations of POS are defined by Chinese Knowledge Information Processing group (CKIP) of Sinica, Taiwan.

Table 1. The parsing rules extracted from the structure tree of Figure 2.

Rule No.	Rules
1	NP←Nhaa
2	VE2←VE2+Dc+VE2
3	NP←DM+Nab
4	S←NP+VE2+NP

Some examples of probabilities of parsing rules are listed in Table 2. We extract 15,946 different parsing transition rules from 90% of Sinica Treebank version 1.0. The other 10% of the structure trees are left for testing.

Table 2. Some examples of parsing rules and their corresponding probabilities.

Parsing rule	Count	Probability
ADV ← A	1	1/1=1
ADV ← Dbaa	4	1/1=1
S ← Cbaa + S	15	15/16=0.9375
VP ← Cbaa + S	1	1/16=0.0625
NP ← NP + A + Nab	5	1/1=1
S ← Cbba + NP + VJ3	1	1/2=0.5
VP ← Cbba + NP + VJ3	1	1/2=0.5
S ← NP + VE12 + NP	7	7/8=0.875
S・的 ← NP + VE12 + NP	1	1/8=0.125
NP ← NP + VG2 + NP	1	1/118=0.008
S ← NP + VG2 + NP	111	111/118=0.941
VP ← NP + VG2 + NP	6	6/118=0.051
S ← NP + VH11 + VC2 + NP	6	1/1=1
S ← Cbca + NP + Dbb + VK2 + NP	1	1/2=0.5
VP ←Cbca + NP + Dbb + VK2 + NP	1	1/2=0.5

5. Determining the POSs of a CFS

We use the 15,946 parsing rules to determine the POSs of a CFS. To accomplish this task, we need a lexicon with POSs for each word. We use ASCED provided by Academia Sinica, Taiwan as the dictionary. ASCED is a well-defined dictionary which contains about 80,000 Mandarin words (中文詞). We also add the possible POSs which defined in ASBC for some words to ASCED. For an input CFS, we first look up the ASCED to get the POSs for each sub-string which is a word in ASCED of the input CFS. And we use these POSs and the 15,946 parsing rules to determine the POS of the input CFS. We try to find out the POSs of a CFS by the POSs of the sub-strings of that CFS. The method we use is a dynamic programming method. Figure 3 is a simple example of determining the POSs of the CFS "林小姐" (Miss Lin).

	1(林)	2(小)	3(姐)
A(林)	Nab, 0.5 Nbc, 0.5	NP, 1	NP, 0.82 Nab, 0.18
B(小)		VH13, 0.25 V3, 0.25 Nv4, 0.25 VH11, 0.25	Nab, 1
C(姐)			b, 1

Figure 3: Determining the POSs of the CFS "林小姐" (Miss Lin).

As shown in Figure 3, we first look up ASCED to find the POSs of each possible word which is a substring of "林小姐". Cell (A,1) contains the possible POSs of the word "林", cell (B,2) contains the possible POSs of "小", cell (C,3) contains the possible POSs of "姐", and cell (B, 3) contains the possible POSs of "小姐". The number following each POS in a cell is the probability of that POS.

We then try to determine the POSs of cell (A, 2) by the parsing rules we extracted from Treebank. The POSs of cell (A, 2)[2] can be derived by the information of cell (A, 1) and cell (B, 2). There are totally 2 * 4 = 8 possible parsing rules derived. By looking at the parsing rules we extracted, we find that only one of the 8 possible combinations exists in the parsing rules. The combination is as follows:

NP ← Nab + Nv4.

(**林小** ← **林** + **小**)

The result of cell (A, 2) is NP. The probability is 1 because Nab + Nv4 can only derive NP. The contents of cell (B, 3) can also be derived from the contents of cells (B, 2) and (C, 3).

We finally determine the POSs of cell (A, 3) via the same method as the preceding step. The POSs of cell (A, 3)[3] can be derived from cells (A, 1) & (B, 3) or cells (A, 2) & (C, 3) or cells (A, 1) & (B, 2) & (C, 3). The results are NP and Nab which are derived from

NP ← Nab + Nab.

(**林小姐** ← **林** + **小姐**)

and

Nab ← Nab + Nab.

(**林小姐** ← **林** + **小姐**)

Finally, the POS of the CFS "**林小姐**" is assigned NP and Nab by inspecting the contents of cell (A, 3). The procedure of assigning the POSs is the bottom-up method of constructing the sentence tree, as shown in Figure 3.

[2] The cell (A,2) represents the substring "**林小**".

[3] The cell (A,3) represents the substring "**林小姐**".

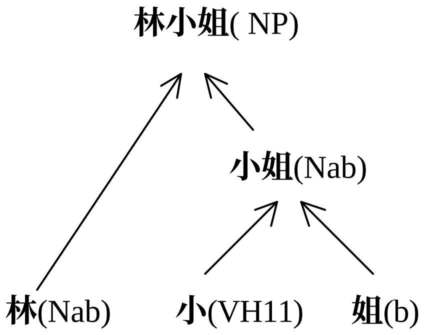

Figure 3: the bottom-up derivation of parsing tree for CFS “林小姐”; NP is assigned finally.

6. The experiment results

The goal of our task is to determine the POSs of CFSs. The testing data we choose are the bottom layer of each structure tree. Each of such testing data contains many words. For example, we will determine the POSs of “**要不要**” and “**這幅畫**” in the example of Figure 2. We found that the POS of “**要不要**” is VE2, and the POS of “**這幅畫**” is NP. We retrieved 1,309 patterns and their related POSs from the testing corpus.

We try to determine the POSs of these 1,309 patterns by our method mentioned in Section 5. The results are shown in Table 3.

Table 3. The accuracy of the 1,309 testing patterns by using all POS tags.

TOP n	**Accuracy**
TOP 1	61.55%
TOP 2	83.87%
TOP 3	91.89%
TOP 4	94.68%
TOP 5	95.82%

The structure of notations of POSs defined by CKIP is a hierarchical structure. There are totally 178 POS tags with five layers in the hierarchical tree [6]. There are 8 categories in the first layer, they are N (noun), C (conjunction), V (verb), A (adjective), D (adverb), P (preposition), I (interjection), and T (auxiliary). The second layer contains 103 POS tags. For example, there are two sub-categories Ca and Cb in the second layer of the category C in the first layer. There are 7 POS tags defined in Sinica Treebank. They are S (sentence), VP (verb phrase), NP (noun phrase), GP (direction phrase), PP (preposition phrase), XP (conjunction phrase), DM (determinate phrase). We also treat these 7 POS tags as in the first layer of the hierarchical tree. If we use the POSs in the first layer, the accuracy of top 1 choice is 72.29%.

Because the size of training corpus is small compared with hundreds of POS tags, we also reduce the tags in each parsing tree to the second layer of the hierarchical tree. For example, when we reduce the POSs tags of the parsing rule

$$S \leftarrow Cbca + NP + Dbb + VK2 + NP$$

to the second layer, we got the reduced parsing rule

$$S \leftarrow Cb + NP + Db + VK + NP.$$

We also determine the POSs of the 1,309 patterns. The result is shown in Table 4. If we use the POSs in the first layer, the accuracy of top 1 choice is 88.53%.

Table 4. The accuracy of the 1,309 testing patterns by using the POS tags in the second layer.

TOP n	**Accuracy**
TOP 1	67.35%
TOP 2	87.74%
TOP 3	95.88%
TOP 4	98.59%
TOP 5	99.46%

Among these 1,309 test patterns, 98 of them are also CFSs. The accuracy for determining the POSs of these 98 CFSs by using all of the POS tags is shown in Table 5. If we use the POSs in the first layer, the accuracy of top 1 is 70.35%.

Table 5. The accuracy of these 98 CFSs by using all POS tags.

TOP n	**Accuracy**
TOP 1	63.26%
TOP 2	78.57%
TOP 3	91.67%
TOP 4	97.62%
TOP 5	97.62%

The reduced parsing rules are also applied to these 98 CFSs. The result is shown in Table 6. If we use the POSs in the first layer, the accuracy of top 1 is 76.28%.

Table 6. The accuracy of the 98 CFSs by using the POS tags in the second layer.

TOP n	**Accuracy**
TOP 1	71.02%
TOP 2	84.53%
TOP 3	92.86%
TOP 4	96.43%
TOP 5	98.81%

We show the defined POSs of some CFSs in Table 7. The gray cell of each row in Table 7 is the correct POS of that CFS in the first column of that row.

Table 7. Some derived POSs of the CFSs. The gray cell is the correct part-of-speech tag of that CFS.

CFS	Top 1	Top 2	Top 3	Top 4	Top 5
說不完	VP				
很快樂	NP	VP	V		
該怎麼說	VP	NP			
這首詩	NP	S	VP		
那位年輕人	NP	S	VP	V	
四年的	NP	NP・之	DM・的	DM・之	
並不容易	VP	S			
大學教授	NP	VP	S		
直接接觸	VP	NP	S	V	
好好讀書	VP	NP	S	V	

7. Conclusions and Future Works

We have proposed a promising method to assign possible POSs to CFSs, adopting the parsing rules extracted from Sinica Treebank. And the precisions of experiments are 71.02% and 98.81% for top 1 and top 5 choices, respectively. The information of POSs can be helpful in many aspects of Chinese natural language processing. In the future, the main work is to expand our method to parse a Chinese sentence, not only assigning the POS for a CFS as described in the paper. The size of Sinica Treebank seems small. If we have a high-performance Chinese sentence parser, we can half-automatically generate more parsing trees for further applications. Another work is to obtain the semantic information, such as “Head” or “negation:” in each sentence of Treebank, and then we can assign the information to a substring of a sentence.

Acknowledgement

We would like to thank Academia Sinica for ASBC corpus, ASCED dictionary, and Sinica Treebank.

References:

1. J. S. Chang, S. D. Chen, S. J. Ker, Y. Chen, and J. Liu, "A Multiple-Corpus Approach to Recognition of Proper Names in Chinese Texts," *Computer Processing of Chinese and Oriental Languages*, Vol. 8, No.1, 1994, pp. 75-85.
2. K. J. Chen and M. H. Bai, "Unknown Word Detection for Chinese by a Corpus-Based Learning Method," *Proceeding of ROCLING X*, 1997, pp.159-174.
3. H. H. Chen and G. W. Bian, "Proper Name Extraction from Web Pages for Finding People in Internet," *Proceeding of ROCLING X*, 1997, pp.143-158.
4. K. J. Chen, C. R. Huang, L. P. Chang, and H. L. Hsu, "SINICA CORPUS: Design Methodology for Balanced Corpora," *Proceeding of PACLIC 11th Conference*, 1996, pp.167~176.
5. H. H. Chen and J. C. Lee, "The Identification of Organization Names in Chinese Texts," *Communication of COLIPS*, Vol.4, No.2, 1994, pp. 131-142.
6. CKIP(Chinese Knowledge Information Processing Group, **詞庫小組**) , "Analysis of Chinese Part-of-Speech (**中文詞類分析**), Technology Report of CKIP #93-05(**中文詞知識庫小組技術報告** #93-05)," Academia Sinica, Taipei, Taiwan, 1993.
7. Eugene Charniak, *Tree-bank Grammars*, AAAI-96, 1996, pp.1031-1036.
8. Keh-Jiann Chen, *A Model for Robust Chinese Parser*, Computa- tional Linguistics and Chinese Language Processing, Vol. 1, No. 1, Taiwan, August 1996, pp. 183-204.
9. Feng-Yi Chen, Pi-Fang Tsai, Keh-Jiann Chen, Chu-Ren Huang, ***中文句結構樹資料庫的構建****(Sinica Treebank)*, Computational Linguistics and Chinese Language Processing, Vol. 4, No. 2, Taiwan, August 1999, pp. 87-104.
10. Walter Daelemans, Antal Van Den Bosch, and Jakub Zavrel, Forgetting Exceptions is Harmful in Language Learning, special issues on natural language learning (34), 1999, pp. 11-34.
11. Y. J. Lin and M. S. Yu, "Extracting Chinese Frequent Strings without a Dictionary from a

Chinese Corpus and Its Applications," *Journal of Information Science and Engineering*, Vol. 17, No. 5, 2001, pp. 805-824.

12. Manny Raner and Samuelsson christer, *Corpus-Based Grammar Specification for Fast Analysis*, *Spoken Language Translator: First Year Report*, SRI Technical report CRC-043, 1994, pp. 41-54.
13. Oliver Streiter, Leonid L. Iomdin, Munpyo Hong, and Ute Hauk, *Learning for Getting and Remembering: Statistical Support for Rule-Based MT*, in TMI, 1999.
14. Oliver Streiter, Hsueh Pie-Yun, *A Case Study on Example-Based Parsing,* ICCL (International Conference on Chinese Language computing), Chicago, USA, 2000.
15. Oliver Streiter, *Parsing Chinese by Examples*, Proceedings of Research on Computational Linguistics Conference (ROCLING), 2000, pp. 111-126.
16. Satoshi Sekine and Ralph Grisman, *Corpus-Based Probability Grammars with only Two Non-terminals*, the 4th International Workshop on parsing Technology, Prague, 1995.
17. M. S. Sun, C. N. Huang, H. Y. Gao, and J. Fang, "Identifying Chinese Names in Unrestricted Texts," *Communication of COLIPS*, Vol.4, No.2, 1994, pp. 113-122.