An LFG Chinese Grammar for Machine Use

Ji Fang and Tracy Holloway King
PARC

Proceedings of the GEAF 2007 Workshop

Tracy Holloway King and Emily M. Bender (Editors)

CSLI Studies in Computational Linguistics ONLINE

Ann Copestake (Series Editor)

2007

CSLI Publications

http://csli-publications.stanford.edu/

Abstract

This paper describes the Chinese grammar developed at PARC, including its three basic components: the tokenizer and tagger, lexicon and syntactic rules. Some of the challenges and issues that we have encountered in the process of development are discussed. In addition, we present our methods of handling these issues. We also illustrate how we evaluate our grammar, providing the evaluation results and some error analyses.

1 Background Introduction

This paper describes a Chinese grammar developed at the Palo Alto Research Center (PARC). This grammar is designed for machine use and is implemented in the framework of Lexical Functional Grammar (LFG) (Kaplan and Bresnan, 1982; Dalrymple, 2001; Bresnan, 2001).

LFG is characterized by its two parallel levels of syntactic representation: Constituent Structure (c-structure) and Functional Structure (f-structure). C-structure encodes information about phrasal structure and linear word order. F-structure encodes information about 'the various functional relations between parts of sentences, information like what is the subject and what is the predicate' (Sells, 1985). Both c-structure information and f-structure information are carried in syntactic rules such as (1).

(1) S –> NP: (^ SUBJ) = !;
VP: ^ = !.

The ^ refers to the f-structure of the mother node and the ! refers to the f-structure of the node itself. (^SUBJ) = ! means that the SUBJ part of the mother's f-structure (the f-structure of the S in (1)) is the f-structure of the node itself (the f-structure of the NP in (1)). ^= ! means that the f-structure of the node itself (the VP in (1)) goes into the f-structure of its mother node (the S in (1)); that is, VP is the functional head of S.

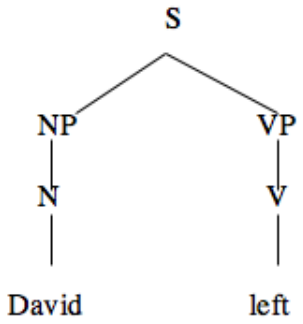
(2) shows two additional phrase structure rules. Together with the rules in (1), these will derive the c-structure and f-structure in (4) and (5) for example (3).

(2) NP –> N: ^ = !;
VP –> V: ^ = !.

[†]Fuji Xerox funded the initial research on the Chinese grammar described in this paper, and we are especially grateful to the support we have received from Tomoko Ohkuma and Hiroshi Masuichi of Fuji Xerox throughout the development. Professor Bing Swen and Professor Shiwen Yu of Beijing University have provided substantial support for the tokenizer and tagger used in this grammar. We also appreciate the feedback they provided during our conversations regarding ways to improve the tokenizer and tagger. We would also like to thank Yuqing Guo from DCU for her work in developing the gold analyses for the 200 gold sentences against which we evaluate our grammar. We also owe our thanks to Emily M. Bender for her helpful feedback and comments on this paper.

(3) David left.

(4) c-structure of (3)



(5) f-structure of (3)

$$\begin{bmatrix} \text{PRED} & \text{'leave<SUBJ>'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{'DAVID'} \end{bmatrix} \end{bmatrix}$$

'leave <SUBJ>' in (5) means that the lexical item 'leave' subcategorizes for a SUBJ. This information comes from the lexicon portion of the grammar.

PARC has been involved in the Parallel Grammar (ParGram) project, which is a world-wide collaborative effort that aims to produce robust and large scale grammars for a wide variety of languages, such as English, German, Japanese, Turkish and Arabic (Butt et al., 1999, 2002). All of these grammars are written within the LFG framework and are implemented on the XLE system (Crouch et al., 2006; Maxwell and Kaplan, 1996) developed by PARC. The Chinese grammar described in this paper is part of the ParGram project.

2 The Chinese Grammar Developed at PARC

Like other grammars in the ParGram project, PARC's Chinese grammar is developed on the XLE system. To parse a sentence, the system minimally requires three types of linguistic specifications: a tokenizer/morphology, a lexicon and syntactic rules. This section describes these three parts of the Chinese grammar.

2.1 Morphology: Segmentation and Tagging

For languages that have morphological inflection such as number, gender, case etc, the morphology processing component of the grammar normally includes a mor-

phological analyzer and a tokenizer.[1] The morphological analyzer also provides part of speech (POS) information for words: therefore, it also functions as a tagger.

In contrast, Chinese is an isolating language, which does not have morphological inflections. Therefore, our grammar does not have a morphological analyzer. Instead, PARC's Chinese grammar uses the tokenizer and tagger developed by Beijing University.[2] The tokenizer and tagger is plugged into the system as a library transducer (Crouch et al., 2006). For an input string such as (6a), the output from the tokenizer and tagger is (6b), which is in turn fed to the XLE system as the input string for syntactic analysis. In order to construct a tree for each lexical item from this type of tagged string, we specify sub-lexical rules such as (7).

(6) a. 窗户开了。
chuāng hu kāi le

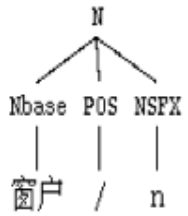
b. 窗户/n 开/v 了/y 。/w
chuānghu kāi le
window open ASP[3] .
'The window is open.'

(7) N –> Nbase
POS
NSFX.

Following (7) and combining information from the lexicon entries for '窗户', '/' and 'n' (as illustrated in (8)), XLE can build a tree and an f-structure for the lexical item "窗户" as shown in (9).

(8) 窗户 Nbase * @ (NON-ANIM-NOUN 窗户).
/ POS * .
n NSFX * (^ CHECK[4]_NSFX)= + .

(9)

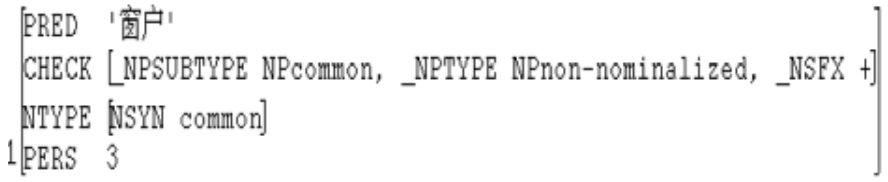


[1]Many languages use finite state morphologies (Beesley and Karttunen, 2003) as part of their XLE grammars (Kaplan et al., 2004a).

[2]http://www.icl.pku.edu.cn/icl_res/

[3]ASP stands for aspect marker.

[4]The CHECK feature is a feature used throughout the ParGram to indicate features that are necessary for internal processing, but not necessary for applications. Applications built on top of the ParGram grammars can delete the CHECK features in their initial processing.



As is broadly acknowledged, Chinese segmentation and tagging are notoriously difficult problems. This is because Chinese does not have morphological inflection, and furthermore, spaces are not inserted between words in written text. For example, the string "有意见" can be segmented as (10a) or (10b), depending on the context.

(10) a. 有 意见
yǒu yìjian
have disagreement

b. 有意 见
yǒuyì jiàn
have the intention meet

The contrast shown in (11) illustrates that even a string that is not ambiguous in terms of segmentation can still be ambiguous in terms of tagging.

(11) a. 白/a 花/n
bái huā
white flower

b. 白/d 花/v
bái huā
in vain spend
'spend (money, time, energy etc.) in vain'

Not surprisingly, the performance of the tokenizer and tagger presents some serious challenges to our grammar as described below.

Challenge #1: Low accuracy of segmentation and tagging Although the tokenizer and tagger developed by Beijing University is state-of-the-art, it achieves about 93% accuracy in segmenting and tagging sentences from the Chinese Treebank5.1 according to our measurements. This level of performance means that each segmented and tagged Chinese sentence of more than 10 words would typically have at least one mistake. Obviously, segmentation and tagging errors directly cause incorrect syntactic analysis and even complete parsing failures.

Challenge #2: Too many verbs In addition to the general accuracy problem, our grammar also suffers from some specific linguistic decisions adopted by the tokenizer and tagger. One such case is illustrated below.

(12) a. 检查/v 病人/n
jiǎnchá bìngren
examine patient

b. 做/v 个/q 检查/v
zuò ge jiǎnchá
do CL examination

In (12), the word 检查 jiǎnchá corresponds to a verb meaning 'examine' in English in (12a), and it corresponds to a noun meaning 'examination' in (12b). Nevertheless, both meanings share the same written form. Zhu (1982, 1985) and Yu (2003) argue that the same word can appear in different syntactic positions and have different grammatical functions; however, that word does not belong to different word categories and should be assigned just one POS tag. Adopting this theory, the tagger developed by Beijing University tags both 检查 jiǎnchá in (12) as a verb. This decision might not be an issue for other tasks or other systems; however, it turns out to be problematic for our grammar.

First, this decision can cause parsing failures. For example, our grammar restricts the category following a classifier in Chinese to be a Noun Phrase (NP), which we believe to be a correct generalization. Following this rule, (12b) will be rejected by the parser because the classifier 个 ge is followed by a verb 检查 jiǎnchá.

This decision also poses an efficiency problem for our grammar. In Chinese, a majority of the verbs have at least two subcategorization possibilities: intransitive and transitive. In the LFG framework, each verb has to satisfy its subcategorization requirements in order to successfully unify. Therefore, putting intransitive and transitive verb entries for everything that is tagged as a verb produces many extra edges in the chart as those verbs try to combine with the words around them as subjects and objects. Consequently, verbs are computationally expensive, and tagging many words as verbs can significantly slow down the parser.

Because our goal is to parse Chinese written text that is not manually segmented or tagged, our grammar implicitly inherits all of the challenges for Chinese segmentation and tagging as well.

Our initial explorations in this area are two-fold. First, we improved the tokenizer and tagger by directly modifying the existing lexical entries and by adding new lexical entries to the dictionary that the tokenizer and tagger use. At the same time, we improve the final segmentation and tagging results by using finite state (FST) rules to post-process the original output from the tokenizer and tagger. For example, a FST rule such as (13) can change an output string from the tokenizer and tagger such as (12b) (repeated below as (14)) to be (15).

(13) v ->n || q "TB" CHARˆ {1,2} "/" _;

(14) 做/v 个/q 检查/v
zuò ge jiǎnchá
do CL examination

(15) 做/v　个/q　检查/n

What (13) specifies is that if a 'v' appears after a '/' following one or two characters, which in turn appear(s) after a 'q' followed by a token boundary (TB), change that 'v' into 'n'. (15) is derived by applying (13) to (14).

In this approach, more information is available in the output string from the tokenizer and tagger compared to the original raw string; by taking into account this additional information, the final segmentation and tagging results are more accurate. For example, compared to the original raw string 做个检查, the output string from the tokenizer and tagger 做/v 个/q 检查/v contains additional information indicating that the string 检查 is preceded by a classifier. With this additional information available, we can safely and accurately change the POS tag of 检查 in this case to be a noun without impacting the tagging of 检查 in other circumstances.

It is also noteworthy that XLE allows non-deterministic segmentation and tagging. In other words, in cases where it is hard to resolve the ambiguity of segmentation or tagging locally, the XLE parser accepts a string with multiple segmentation possibilities and a token with multiple possible tags. For example, the word 选举 xuǎnjǔ 'elect/election' is equally frequently used as a verb and as a noun. In this case, the best solution is to allow both the 'v' and the 'n' tag and hand off the resolution of that ambiguity to the syntactic processor. Similarly, when the ambiguity of segmentation is hard to resolve locally, multiple segmentation results for a string are allowed, and the XLE parser will try all of these different results as input to the grammar.

Based on our initial observations, the system has been improving with the FST rules integrated. However, more work still needs to be done in this area. As part of this process, we plan to evaluate how much the FST rules improve the tokenizer and tagger against grammar performance on real world data.

To summarize, this section describes the tokenizer and tagger that we integrate into our grammar, the challenges that our grammar has to face in this regard and our approach to improve the tokenizer and tagger. The following sections describe the other two important components of the grammar, namely the lexicon and syntactic rules.

2.2 Lexicon

The lexicon component of the grammar contains lexical entries specifying information particular to different lexical items. For example, (16) is the lexical entry for the noun 猫 māo 'cat', and (17) is the lexical entry for the verb 加入 jiārù 'join' in the Chinese grammar.

(16) 猫　　Nbase * @ (ANIM-NOUN 猫).

(17) 加入 Vbase * { @(V-SUBJ 加入)
(^ SUBJ CHECK _SUBJ-TYPE)=c np
|@(V-SUBJ-OBJ 加入)
(^ SUBJ CHECK _SUBJ-TYPE)=c np}.

(16) specifies that the category of the lexical item 猫 māo is Nbase. Combining this entry with information from lexical entries for POS tags produced by the tagger (as shown in (18)) and sub-lexical rules such as (7) (repeated below as (19)), XLE can build a c-structure such as (20) for 猫 māo.

(18) / POS * .
n NSFX * (^ CHECK _NSFX)= +.

(19) N –> Nbase
POS
NSFX.

(20) c-structure of 猫

(16) also invokes a template ANIM-NOUN (shown in (21)), which defines features and values for all animate nouns in Chinese.

(21) ANIM-NOUN(_P) =
(^ PRED)='_P'
(^ PERS)=3
(^ ANIM)=+
(^ NTYPE NSYN)=common
(^ CHECK _NPTYPE)=NPnon-nominalized
(^ CHECK _NPSUBTYPE)=NPcommon

Combining this information, XLE can build an f-structure for 猫, as illustrated in (22).

(22) f-structure of 猫

```
PRED   '猫'
CHECK [_NPSUBTYPE NPcommon, _NPTYPE NPnon-nominalized, _NSFX +]
NTYPE [NSYN common]
ANIM +, PERS 3
```

Similarly, (16) defines 加入 jiārù ‘join’ as a verb that can be used either intransitively or transitively. It also specifies that the subject of 加入 must be a NP.

Currently, the lexicon component of the Chinese grammar has several hundred manually coded lexical entries, including closed class items such as punctuation. We handle words that do not have a listed lexical entry through a “guesser” lexical entry exemplified in (23).

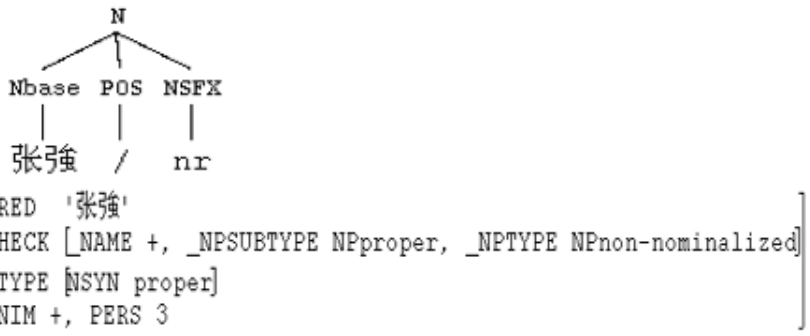
(23) -unknown Nbase * @(PROPER-ANIM-NOUN %stem)
(^ CHECK _NAME)=c +.

(23) specifies that if a lexical item does not have a lexical entry elsewhere, it can be posited as an Nbase; if it has a feature ‘CHECK _NAME’ whose value is ‘+’, then it is an animate proper noun. The value ‘+’ of ‘CHECK _NAME’ is derived from the POS tag produced by the tokenizer and tagger: the tagger tags a person’s name as ‘nr’, and we assign ‘(^ CHECK _NAME) = +’ in the lexical entry of ‘nr’, as shown in (24).

(24) nr NSFX * (^ CHECK _NAME)= +.

Combining information from (23), (24) and the template of PROPER-ANIM-NOUN, the c-structure and f-structure of a person’s name such as 张强 zhāngqiáng ‘Zhang, Qiang’ are illustrated in (25).

(25)



In addition to names, our guesser postulates locative, time and common nouns, as well as adjectives, adverbs, numbers, classifiers, conjuncts, prepositions and verbs in a similar way: we first write a lexical entry for each tag, such as (24) for the ‘nr’ tag; we also assign a value to a feature for each tag, for instance, ‘(^ CHECK _NAME)= +’ is assigned for the tag ‘nr’ (as in (23)), and ‘(^ CHECK _VSFX)= +’ is assigned for the tag ‘v’. The guesser then posits the category of the unknown item based on the ability to form a particular c-structure category via the sublexical rules and on the value of particular features. This process is quite efficient in part because XLE first builds the c-structure, before any unification occurs, and hence many possible entries are eliminated early in the parsing process. For example, if the f-structure of the unknown item contains a feature ‘(^ CHECK _VSFX)’ whose value is ‘+’, that item must be associated with a tag ‘v’, thus the guesser

can postulate the unknown item as a verb. Note that because the tags and the '/' do not have PREDs, their features and values are projected to the mother node's f-structure, which is identical to the f-structure of the unknown lexical item.

Verbs pose the biggest challenge to the guesser. In LFG, subcategorization information is required for verbs. However, this information is not encoded in the 'v' tag of verbs returned by the tagger, and we have not found any suitable resource from which we can extract the subcategorization requirements for verbs in Chinese. We have manually coded the entries for some high frequency verbs.[5] These verbs do not go through the *-unknown* entry. For all other verbs, our compromise solution is to postulate each unknown verb to be either intransitive or transitive. The guesser also allows a verb to subcategorize for a XCOMP, if the PRED form of the XCOMP is the PRED form of one of the verbs, such as 为wéi 'be'.[6]

Nevertheless, the lack of reliable and complete subcategorization information for Chinese verbs poses challenges for our grammar, as discussed in the evaluation section of this paper. Possible enhancements are discussed in section 4.

2.3 Syntactic Rules

The third part of the grammar involves the Chinese syntactic rules. Currently the grammar has 114 rules with 2203 states and 4301 arcs.[7] This means that the grammar has 114 left-hand side categories (such as the X in 'X -> Y Z') in its phrase-structure rules, and these 114 rules compile into a collection of finite-state machines with 2203 states and 4301 arcs (Butt et al., 2002). The grammar covers common phrasal constructions such as NPs, VPs, ADJPs, PPs and ADVPs. The grammar also covers all four clause types in Chinese: declarative, interrogative, exclamatory and imperative.

Due in part to the lack of morphology, Chinese tends to present many ambiguities at both the c-structure and f-structure level. For example, for a NP such as (26), the internal NP structure can be very ambiguous (5 trees), as shown in (27).

(26)	国民	生产	总值
	guómín	shēngchǎn	zǒngzhí
	people	produce	total value
	'GDP'		

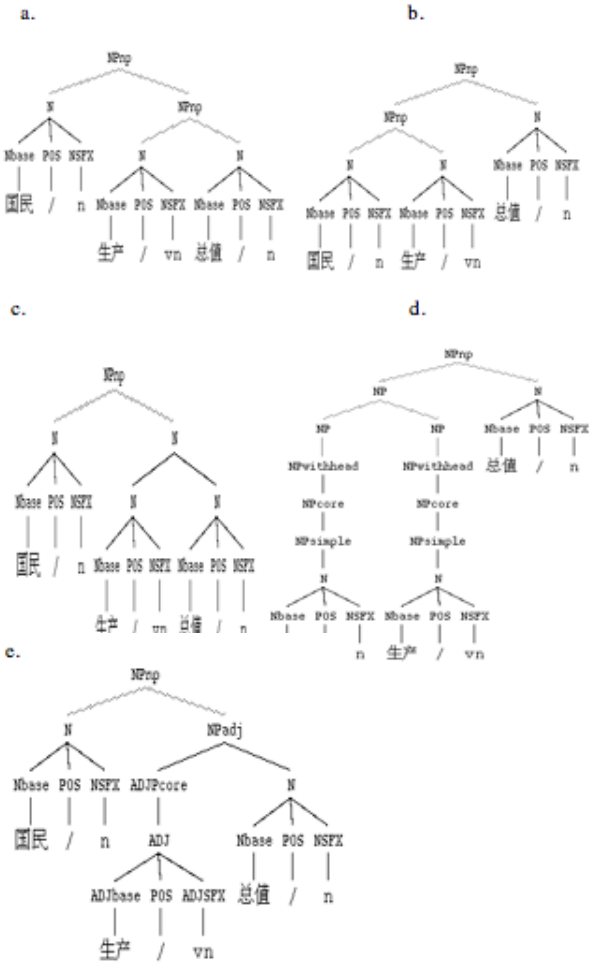
[5]Currently the grammar has manually encoded entries for the 20 most frequent verbs, and our goal is to expand the lexicon to contain entries for the 100 most frequent verbs in Chinese.

[6]Verbs such as 为 wéi 'be' are frequently used as the 补语 bǔyǔ ('complement' in conventional Chinese linguistic terminology) in a 动补结构 dòng bǔ jiégòu 'verb-complement construction' such as 变为 biàn wéi 'change to be'. 补语 bǔyǔ in the 动补结构 dòng bǔ jiégòu is treated as XCOMP in our grammar.

Some common XCOMP verbs are 完wán 'complete', 尽jìn 'complete', 成chéng 'succeed', 上来shànglai 'go up (towards the speaker)' , 上去shàngqu 'go up (away from the speaker)', 下来xiàlai 'go down (towards the speaker)', 下去xiàqu 'go up (away from the speaker)', 起来qǐlai 'get up/begin', 起qǐ 'get up/begin', 去qù 'go', 来lái 'come', 出来chūlai 'come out', 出去chūqu 'go out' and 回来huílai 'come back'.

[7]XLE compiles the grammar into a finite-state machine.

(27)



The corresponding f-structures of these c-structures are also different: while (27c) and (27d) involve coordination, the others not.

The ambiguity issue is one of the contributing factors to the current grammar's efficiency issues. The following section describes this problem in a greater detail.

3 Evaluation and Error Analysis

To evaluate the coverage and accuracy of the grammar (Crouch et al., 2002; Kaplan et al., 2004b), we use a set of 200 sentences from the CTB5 (Xue et al., 2002, 2005) chosen by Dublin City University (DCU) as gold standard sentences for evaluating Chinese deep grammars. These 200 sentences are 10–20 words long. 50.5% of the sentences are chosen from the Xinhua sources, 3.5% are from HKSAR and 46% are from Sinorama. The topics of the gold standard sentences cover economics, politics, culture, sports and entertainment. The writing style of the Xinhua and HKSAR sentences is formal, whereas the writing style of the Sinorama sentences

are mixed: some are formal, and some are colloquial. All 200 sentences were unseen to the grammar prior to the evaluation.

We also use DCU's gold analysis as a basis for evaluating the accuracy of the grammar. However, PARC's analysis is based on the segmentation and tagging results from the integrated tokenizer and tagger, which are different from the segmentation and tagging in CTB5 on which the DCU gold standard analysis is based. Therefore, some of the errors in the results shown below are caused by the different segmentation and tagging standards adopted by CTB5 and the tokenizer and tagger developed by Beijing University. The reason why we did not evaluate our grammar based on gold segmented and tagged sentences is because we want to know how good the results would be over novel, untokenized text.

Parsing the 200 gold standard sentences with PARC's Chinese grammar (as of March, 2007), 188 sentences had full parses, 7 sentences had fragmented parses, 4 sentences ran out of storage (the maximum storage is set as 1500 MB in this grammar), and 1 sentence had 0 parse, as shown in Table 1. Fragmentation Rate is 3.5%.

Table 1: Coverage Results

Total	Full Parses	Fragmented Parses	Ran Out of Storage	No Parse
200	188	7	4	1

To evaluate the accuracy, we adopt the same algorithm described in Crouch et al. (2002). The results are shown in Table 2.

Table 2: Accuracy Results

TOTAL:	precision=73.1	recall=72.4	f-score=72.7
DEPENDENCY	PRECISION	RECALL	FSCORE
adjunct	939/1228 = 76	939/1267 = 74	75
comp	15/22 = 68	15/25 = 60	64
conj	182/271 = 67	182/278 = 65	66
obj	330/452 = 73	330/449 = 73	73
obl_ag	9/9 = 100	9/9 = 100	100
passive	9/9 = 100	9/9 = 100	100
subj	318/478 = 67	318/455 = 70	68
topic	0/0 = 0	0/1 = 0	0
xcomp	42/53 = 79	42/55 = 76	78

The Chinese gold standard has only predicate argument/adjunct structure (that is, everything with a PRED and the path into it). There are no 'easy' features like

CLAUSE-TYPE, V-TYPE, PERS, which tend to be correct if the core structure is correct. Therefore, the f-score would likely be higher if the Chinese gold standard did contain those features.

As mentioned above, some of the mismatches between the analyses produced by PARC's Chinese grammar and DCU's gold analyses are caused by the different segmentation and tagging standards adopted by CTB5 and the tokenizer and tagger that our grammar uses. For example, CTB5 treats 第 dì 'ordinal marker' + number as one word, whereas the tokenizer used in our grammar treats it as two separate words.

The tagging standard between the two systems is also different. For example, for the same string in (28), (29a) is the tagging results from the tagger in our grammar, and (29b) is the tagging result from CTB5.

(28)

证券	市场	健康	发展	的
zhèngquàn	shìchǎng	jiànkāng	fāzhǎn	de
stock	market	healthily/ health	develop/ development	MM[8]
重大	举措			
zhòngdà	jǔcuò			
important	measure			

(29) a. 证券/n 市场/n 健康/a 发展/v 的/u 重大/a 举措/n

b. 证券_NN 市场_NN 健康_JJ 发展_NN 的_DEG 重大_JJ 举措_NN

The major difference between (29a) and (29b) is that in (29a), the word 发展 fāzhǎn is tagged as a verb, meaning 'develop', while in (29b), it is tagged as a noun, meaning 'development'. At first glance, (29b) seems to yield a reading of 'the important measure regarding the healthy development of the stock market', which is very parallel to the English structure. However, (29b) cannot yield such a reading: this is because if 发展 fāzhǎn is a noun, and its adjunct is 健康 jiànkāng, we should be able to insert a 的 de, which introduces a head NP, rather than a 地 de, which introduces a head VP, between them. However, 健康的发展 jiànkāngdefāzhǎn only entails 'the development of health' in Chinese; in contrast, 健康地发展 jiànkāngdefāzhǎn can entail 'healthy development (the nominalization of 'develop healthily')'. Therefore, while (29a) would yield a reading of 'the important measure (which assures that) the stock market develops healthily' or 'the important measure (which assures) the healthy development of the stock market', (29b) would mean 'the important measure regarding the development of the stock market's health'. Such a difference would yield very different analyses. Based on (29a), 证券市场 zhèngquàn shìchǎng 'stock market' is the subject of 发展 fāzhǎn 'develop'; while based on (29b), 证券市场 zhèngquàn shìchǎng 'stock market' is an adjunct.

[8]MM stands for modifier marker.

In addition to this mismatch, another significant source of errors is our incomplete lexicon resources. In the 200 gold sentences, 17 sentences receive incorrect analyses for this reason. Among the 17 sentences, five fail due to the lack of proper subcategorization information for verbs, while the remaining 12 fail due to missing lexical entries for other lexical items.

Chinese tends to be ambiguous in both the c-structure and f-structure levels as described above. One way to control ambiguity is to use a special optimality (OT) mark (Frank et al., 2001) called a STOPPOINT provided by the XLE system. In XLE, if an analysis contains an OT mark that is ranked behind the STOPPOINT, that analysis is not tried unless everything else fails. Therefore STOPPOINT is useful for eliminating rare and incorrect analyses when correct analyses are present and for speeding up the parser in those cases.

Ironically, although Chinese has been recognized as a topic prominent language (Li and Thompson, 1981), we place the topic analysis before the STOPPOINT, because we have observed that allowing the topic analysis significantly slows down the parsing while not greatly increasing accuracy. The following three reasons are likely to be responsible for the inefficiency caused by including the topic analysis in the grammar. First, the position of topics in Chinese is flexible. A topic can occur in the first or second NP position, and a sentence can have more than one topic, as demonstrated in (30).

(30) a. 苹果 我 喜欢。
píngguǒ wǒ xǐhuān
apple I like
'Apples, I like.'

b. 大 城市 北京 我 最 熟悉。
dà chéngshì běijīng wǒ zuì shúxī
big city Beijing I most familiar
'Among big cities, I am most familiar with Beijing.'

(30a) has one topic, which is 苹果 píngguǒ 'apple' that appears in the first NP position; (30b) has two topics. While the external topic 大城市 dà chéngshì 'big cities' occurs in the first NP position, the internal topic 北京 běijīng 'Beijing' occurs in the second NP position. Second, unlike the topic in English, which must be linked to another grammatical function, the topic in Chinese is not necessarily linked to any another grammatical function. For example, while the topic in (30a) is linked to the grammatical function of object, the topics in (30b) are not linked to any other grammatical function. Third, topics generally occur in a 'NP1 NP2' sequence at the sentence-initial position; however, it is very common to analyze NP1 as the adjunct, possessor or conjunct of NP2 in a 'NP1 NP2' sequence. Therefore, allowing topic analyses significantly increases the level of ambiguity for sentence initial 'NP1 NP2' sequences, which are extremely common in Chinese sentences according to our observation.

At the same time, the topic function often overlaps with other grammatical functions such as adjunct in Chinese. For example, in (31), 他 tā 'he' can both be understood as the topic of the entire sentence or the adjunct of 肚子 dùzi 'stomach'. Therefore, it does not seem to be a significant drawback if the topic analysis is blocked unless it is the only possible analysis.

(31) 他 肚子 饿。
tā dùzi è
he stomach hungry
'He is hungry.'

By placing the topic analyses behind the STOPPOINT, the grammar's efficiency is improved. However, occasionally, the intended topic analysis will be suboptimal and hence not available. In the 200 gold sentences, one sentence should have a topic analysis that is incorrectly suppressed by our system as shown in the accuracy results above.

Another method that we adopt to control ambiguity is to use OT marks more generally to rank preferences for different analyses. Through this method, the less common analyses can be suppressed as suboptimal analyses. All of the OT marks in the Chinese grammar are manually coded, and it is noteworthy that a significant number (24 out of the 200 sentences) of correct analyses are incorrectly suppressed as suboptimal analysis by the OT marks specified in the grammar. The suppressed suboptimal analyses cannot be picked to compare against the gold standard, which implies that the OT marks in the Chinese grammar need to be better tuned in order to improve the grammar's performance.

4 Summary and Future Work

This paper describes the Chinese grammar developed at PARC, including its three basic components, namely, the morphology, lexicon and syntactic rules. We also describe the challenges and issues that we have encountered in the process of development, as well as our methods of handling these issues. In addition, we illustrate how we evaluate our grammar, including the evaluation results and some error analysis.

The three major challenges currently confronted by our grammar are (1) the tokenizer and tagger; (2) lexicon resources such as subcategorization requirements of verbs; and (3) ambiguity control.

As far as the tokenizer and tagger is concerned, the initial results of improvements to segmentation and tagging accuracy by using FST patch rules to post-process the original results returned by the tokenizer and tagger are encouraging. We will continue our investigations in this direction and plan to investigate integrating machine learning algorithms in this process.

Because the subcategorization information of verbs is critical to our system, we are looking for suitable resources from which we can automatically extract this

information. Resources such as Chinese Word Net or electronic verb dictionaries can be useful. We are also considering learning the subcategorization information from the Chinese Treebank.

C-structure pruning (Crouch et al., 2006) has proven to be very effective in terms of reducing ambiguity and accelerating the parser for the English grammar developed at PARC and the German grammar developed at the University of Stuttgart. We expect that this technique can help mitigate the ambiguity issue of our Chinese grammar as well.

Despite all of the challenges, the Chinese grammar described in this paper has reached a relatively stable stage, and we are planning to use it as a base to produce Chinese core semantics parallel to that developed for English (Crouch and King, 2006). We also plan to use this grammar to start initial exploration on Chinese-English and English-Chinese machine translation.

References

Beesley, Kenneth R. and Karttunen, Lauri. 2003. *Finite State Morphology*. CSLI Publications.

Bresnan, Joan. 2001. *Lexical-Functional Syntax*. Blackwell Publishers.

Butt, Miriam, Dyvik, Helge, King, Tracy Holloway, Masuichi, Hiroshi and Rohrer, Christian. 2002. The Parallel Grammar Project. In *Proceedings of COLING-2002 Workshop on Grammar Engineering and Evaluation*, pages 1–7.

Butt, Miriam, King, Tracy Holloway, Niño, María-Eugenia and Segond, Frédérique. 1999. *A Grammar Writer's Cookbook*. CSLI Publications.

Crouch, Dick, Dalrymple, Mary, Kaplan, Ron, King, Tracy Holloway, Maxwell, John and Newman, Paula. 2006. XLE documentation, http://www2.parc.com/isl/groups/nltt/xle/doc/.

Crouch, Dick and King, Tracy Holloway. 2006. Semantics via F-Structure Rewriting. In *Proceedings of LFG06, CSLI On-line publications, pp. 145-165.*.

Crouch, Richard S., abd Tracy Holloway King, Ronald Kaplan and Riezler, Stefan. 2002. A comparison of evaluation metrics for a broad coverage parser. In *Beyond PARSEVAL – Towards Improved Evaluation Measures for Parsing Systems: LREC 2002 Workshop*.

Dalrymple, Mary. 2001. *Syntax and Semantics. Volume 34: Lexical Functional Grammar*. Academic Press.

Frank, Anette, King, Tracy Holloway, Kuhn, Jonas and Maxwell, John T. 2001. Optimality Theory Style Constraint Ranking in Large-scale LFG Grammars. In Peter Sells (ed.), *Formal and Empirical Issues in Optimality Theoretic Syntax*, pages 367–397, CSLI Publications.

Kaplan, Ron, Maxwell, John T., King, Tracy Holloway and Crouch, Richard. 2004a. Integrating Finite-state Technology with Deep LFG Grammars. In *Proceedings of the Workshop on Combining Shallow and Deep Processing for NLP (ESSLLI)*.

Kaplan, Ron, Riezler, Stefan, King, Tracy Holloway, Maxwell, John T., Vasserman, Alex and Crouch, Richard. 2004b. Speed and Accuracy in Shallow and Deep Stochastic Parsing. In *Proceedings of HLT-NAACL'04*.

Kaplan, Ronald. M and Bresnan, Joan. 1982. Lexical-Functional Grammar: A formal system for grammatical representation. In Joan Bresnan (ed.), *The Mental Representation of Grammatical Relations*, pages 173–281, The MIT Press.

Li, Charles N. and Thompson, Sandra A. 1981. *Mandarin Chinese: A Functional Reference Grammar*. University of California Press.

Maxwell, John and Kaplan, Ron. 1996. An Efficient Parser for LFG. In *Proceedings of the First LFG Conference*, CSLI Publications.

Sells, Peter. 1985. *Lectures on Contemporary Syntactic Theories*. CSLI Publications.

Xue, Nianwen, Chiou, Fu-Dong and Palmer, Martha. 2002. Building a Large-Scale Annotated Chinese Corpus. In *Proceedings of the 19th. International Conference on Computational Linguistics*.

Xue, Nianwen, Xia, Fei, Chiou, Fu-Dong and Palmer, Martha. 2005. The Penn Chinese TreeBank: Phrase Structure Annotation of a Large Corpus. *Natural Language Engineering* pages 207–238.

Yu, Shiwen et al. 2003. *The Grammatical Knowledge-base of Contemporary Chinese — A Complete Specification*. Qinghua University Press.

Zhu, Dexi. 1982. *Yufa Jiangyi (Lectures on Grammar)*. Shangwu Yinshuguan.

Zhu, Dexi. 1985. *Yufa Dawen (Questions and Answers Regarding Chinese Grammar)*. Shangwu Yinshuguan.