Chinese Treebanks and Grammar Extraction

Keh-Jiann Chen
Institute of Information Science,
Academia Sinica, Taipei
kchen@iis.sinica.edu.tw

Yu-Ming Hsieh
Institute of Information Science,
Academia Sinica, Taipei
morris@iis.sinica.edu.tw

Abstract

Preparation of knowledge bank is a very difficult task. In this paper, we discuss the knowledge extraction from the manually examined Sinica Treebank. Categorical information, word-to-word relation, word collocations, new syntactic patterns and sentence structures are obtained. A searching system for Chinese sentence structure was developed in this study. By using pre-extracted data and SQL commands, the system replies the user's queries efficiently. We also analyze the extracted grammars to study the tradeoffs between the granularity of the grammar rules and their coverage as well as ambiguities. It provides the information of knowing how large a treebank is sufficient for the purpose of grammar extraction. Finally, we also analyze the tradeoffs between grammar coverage and ambiguity by parsing results from the grammar rules of different granularity.

Key Words: treebanks, knowledge extraction, grammar coverage, ambiguities, parsing.

1 Introduction

Parsing natural language sentences makes use of many different knowledge sources, such as lexical, syntax, semantic, and common sense knowledge (Chen, 1996a; Pustejovsky, 1985). Preparation of knowledge bank is a very difficult task, since there are vast amount of knowledge and they are not well organized (Tseng et al., 1988). The Corpus-based approach provided a way of automatically extract different knowledge. From part-of-speech tagged corpora (Chen et al., 1994a; Chen et al., 1996b; CKIP, 1993) to the syntactic structure annotated treebanks (Marcus et al., 1993), each contributes explicit linguistic knowledge at different level for better automation on knowledge extraction. Treebanks provide an easy way for extracting grammar rules and their occurrence probability. In addition, word-to-word relations (Chen 1992; Pollard et al., 1994) are also precisely associated. Hence it raises the following important issues. How will treebanks be used? How many annotated tree structures are sufficient in a treebank for the purpose of grammar generation? What are tradeoffs between grammar coverage and ambiguities? We will try to answer the above questions in the following sections.

1.1 Introduction to Sinica Treebank

Sinica Treebank has been developed and released to public since 2000 by Chinese Knowledge Information Processing (CKIP) group at Academia Sinica. Sinica Treebank version 2.0 (9 files) contains 38944 structural trees and 240,979 words in Chinese. Each structural tree is annotated with words, part-of-speech of words, syntactic structure brackets, and thematic roles. For conventional structural trees, only syntactic information was annotated. However, it is very important and yet difficult for Chinese to identify word relations with purely syntactic constraints (Xia et al., 2000). On the other hand, a purely semantic approach has never been attempted for theoretical and practical considerations (Chen et al., 2000). Thus, partial semantic information was annotated in our Chinese structural trees. That is, grammatical constraints

are expressed in terms of linear order of thematic roles and their syntactic and semantic restrictions.

Ta jiao Li-si jian qiu.
He ask Lisi pick ball.
"He asked Lisi to pick up the ball."

S(agent:NP(Head:Nhaa:Ta'He')|
Head:VF2:jiao'ask'|goal:NP(Head:Nba:Li-si)|
theme:VP(Head: VC2: jian 'pick'|
goal:NP(Head:Nab:qui'ball')))

Figure 1. An example

The representation of the dependency tree, as in Figure 1, has the advantages of maintaining phrase structure rules as well as the syntactic and semantic dependency relations (Chen et al., 1994b). The meaning of each grammatical category is listed in the Appendix 1.

2 Uses of treebanks and grammar extraction

Here we intend to find the useful information behind Sinica Treebank and transfer it into a formatted knowledge that the language analyzer can use.

2.1 Knowledge extraction from treebanks

From Sinica Treebank, four different types of information were extracted. They are a) Lexical and categorical information, b) Word-to-Word relations, c) Word Bi-grams, and d) Grammar rules.

A searching system of using the above four information has been developed. Users can use this searching system via a web browser at http://140.109.19.103/treesearch/. The searching system architecture is shown in Figure 2.

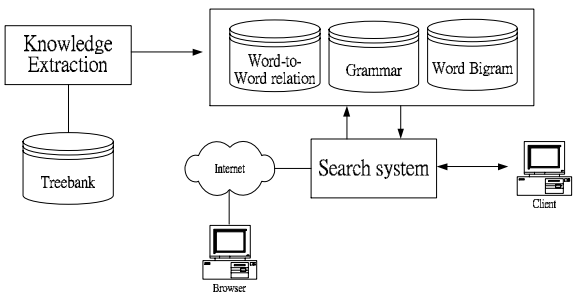

Figure 2.The Tree-Searching system

The system provides the users with "Keyword Search" and "Sentence structure search" functions. The system can perform filtering and aggregating on the searching results.

By using the Treebank Searching System, we also found some annotation errors in the original treebank. Such information can be obtained from the statistical data of syntactic category and role. Therefore, the original content of the trees were corrected to improve the quality of the annotation.

2.2 Uses of the extracted information

Text annotation is for the purpose of making implicit knowledge in documents more explicit. Structure annotations, in particular, make the grammatical relations of a sentence explicit. The uses of each type of the extracted information from treebank are exemplified below.

Supposed that we want to know what grammatical functions of a syntactic category are, say VC2 (active transitive verb). We can search for the lexical/categorical data and get the results of Table 1. It shows that the active transitive verbs (VC2) will play the role of sentential head mostly. They occurred 8389 times in the treebank. The verb VC2 also functions as modifier of noun (property role), predication of a relative clause, and surprisingly adverbial manner role. The roles of DUMMY are conjuncts of conjunctive constructions.

Role	Frequency
Head	8389
DUMMY1	27
DUMMY2	26
property	10
predication	10
manner	10

Table 1. The thematic roles played by the verb type VC2

The extracted word-to-word relations are mostly head-modifier and head-argument relations, which are also instances of world knowledge. For example, we can extract the knowledge of what entities are eatable from the argument of the verb 'eat'. Collocations are very useful information for lexicography and NLP. If we sort the extracted word-to-word relations, the most frequent relations are

listed in Table 2. We also find some interesting linguistic patterns uniquely for Chinese language.

Left word	Right word	Frequency
在(zai)	中(zhong)	348
在(zai)	上(shang)	318
是(shi)	的(de)	201
是(shi)	就(jiu)	183
是(shi)	這(zhe)	150
是(shi)	也(ye)	145

Table 2. Some common collocations found by word-to-word relations

Word bi-gram statistics is often the major information for constructing language model (Yuan et al., 1997; Manning et al., 1999). Some other interesting information can also be extracted. For instance, to identify personal names in Chinese text their context information is very useful. Following table shows the collocations of proper names and most of them are titles of people.

Category	Word	Frequency
DE	的(de)	373
P21	在(zai)	132
VE2	表示(biao shi)	95
Cab	等(deng)	86
Caa	和(he)	79
…	…	…
Nab	總統(zong tong)	43
Nab	教練(jiao lian)	25
Nab	選手(xuan shou)	24
Nab	外長(wai zhang)	20
Nab	主席(zhu xi)	18

Table 3. Words frequently co-occurred with proper names

Grammar rule extraction is the major usage of Treebanks (Uszkoreit, 1986). Not only sentential/phrasal patterns but also their probabilities of usages can be derived as exemplified in Table 4. The probabilistic context-free grammars are proven to be very effective for parsing natural languages (Gazdar et al., 1985).

Rule	Freq.
Head-VC2 goal-NP	1713
Head-VC2	629
agent-NP Head-VC2 goal-NP	316
Head-VC2 goal-NP complement-VP	190
agent-NP Head-VC2	153
time-Dd Head-VC2 goal-NP	105

Table 4. The top 6 high frequency sentential patterns of the active transitive verb (VC2)

3 Grammar coverage and ambiguities

One of the major purposes of construction of treebanks is for grammar extraction. Probabilistic phrase structure rules can be derived from treebanks. However how many annotated tree structures are sufficient for the purpose of grammar generation? What are tradeoffs between the granularity of grammar representation and grammar coverage as well as ambiguities? We try to answer the above questions in this section.

3.1 Granularity vs. grammar coverage

In order to see how the size of treebank affects the quality of the grammar extraction, we use treebanks in different sizes and in different levels of granularities to extract grammars and then compare their coverage and ambiguous rates. The four levels of grammar representations are from fine-grain representation to coarse-grain representation. For example, the extracted lexical units and grammar rules of the tree in Figure 1 are listed as follows. At fine-grain level each lexical unit is a thematic role constraint by the word and its phrasal category. Each rule is represented by a sequence of lexical/categorical units. At the three lower level representations, the lexical units are syntactic category based. The set of categories are from Case-2 fine-grain categories to Case-4 coarse-grain categories. Each lexical unit is a thematic role constraint by the lexical category and phrasal category. See Appendix 2 for category mapping between different levels.

Case-1: Fine-grain level (Word-Level)
S(agent:NP()|jiao|goal:NP()|theme:VP()),
agent:NP(Ta),
goal:NP(Li-si),
theme:VP(jian|goal:NP()),
goal:NP(qiu)

Case-2: Category level
S(agent:NP()|VF2|goal:NP()|theme:VP()),
agent:NP(Nhaa),
goal:NP(Nba),
theme:VP(VC2|goal:NP()),
goal:NP(Nab)

Case-3: Simplified-Category level
S(agent:NP()|VF|goal:NP()|theme:VP()),
agent:NP(Nh),
goal:NP(Nb),
theme:VP(VC|goal:NP()),
goal:NP(Na)

Case-4: Coarse-grain level
S(agent:NP()|V|goal:NP()|theme:VP()),
agent:NP(N),
goal:NP(N),
theme:VP(V|goal:NP()),
goal:NP(N)

It is clear that fine-grain grammar representation would have less grammar representational ambiguity, but with lower grammar coverage. On the other hand, the coarse-grain grammar representation is more ambiguous but with better coverage. The experiments were carried out to show the above-mentioned tradeoffs.

In order to answer the question of how many annotated tree structures are sufficient for the purpose of grammar generation, the grammar extraction processes were carried out on the treebanks of four different sizes, each with 10000, 20000, 30000, and 38725 trees. We exam the grammar coverage of each set of grammar rules extracted from the treebanks of different sizes. For each treebank, we divide the treebank into ten equal parts. For example, we obtain $db_1^1 ... db_1^{10}$ from the treebank db_1 of size 10000 trees. Each part has 1000 trees. The grammar coverage was estimated as follows. For each part, we analyze its coverage rate by the grammar extracted from other 9 parts and average 10 coverage rates to be the coverage rate of the grammar derived from the experimental treebank. The grammar coverage experiments were carried out for all four different sizes of treebanks and for four different levels of granularities. The results are shown in Table 5 and depicted in Figure 3.

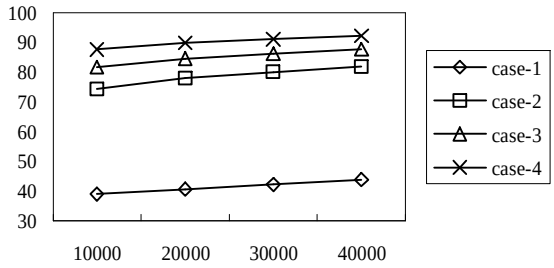

Figure 3. Coverage rates vs. size of treebanks

Size / Level	10000	20000	30000	40000
Case-1	38.97%	40.67%	42.25%	43.95%
Case-2	74.28%	78.15%	80.08%	81.91%
Case-3	81.75%	84.56%	86.31%	87.71%
Case-4	87.67%	89.92%	91.14%	92.2%

Table 5. Grammar coverage rates

The results indicate that as we expected the fine-grain rules have the least coverage rate, while the coarse-grain rules have the highest coverage rate. The coverage rate increases when the size of treebank increases. Since they are not in linear proportion, it is hard to predict exactly how large amount of trees are required in order to derive grammar rules with sufficient coverage. However, the result did show us that the size of current treebank is not large enough to derive a fine-grain rule set with high coverage rate. Only the coarse-grain rules can reach up to 92.2% coverage rate, but the coarse-grain rules suffer from high ambiguity rates.

3.2 Granularity vs. ambiguities

We intend to measure the ambiguity rate of a set of grammar rules from parsing point of view. A parsing process needs to decide the thematic role of each lexical token and decide which rules to apply. Therefore a simple way of measuring ambiguity of a grammar representation is to see how many possible thematic roles for each lexical item may played and how many different rules contains this token in the grammar representation. We consider four levels of granularities as defined in the above section. The lexical item for four levels of granu-

larity are ”Category:Word”, “Category”, “Simplified-Category”, and “Coarse-grain Category” respectively. We use the grammar extracted from the whole treebank as the target of investigation. For four different levels of granularities, Table 6 shows the number of ambiguous roles in average played by each lexical item and the average number of grammatical rules partially matched a particular lexical item. The results did support the claim that fine-grain grammar representation would have less grammar representational ambiguity and the coarse-grain grammar representation is more ambiguous but with better coverage.

Event / Level	# of lexical items	Role ambiguities	# of grammatical rules	Rule ambiguities
1	38,927	1.19	82,221	2.69
2	190	3.08	24,111	132.47
3	47	5.23	15,788	350.84
4	12	9.06	10,024	835.30

1: Category:Word, 2: Category,
3: Simplified-Category, 4: Coarse-grain Category

Table 6. Role ambiguities of the lexical item of the form Category:Word

4 Parsing Results

The probabilistic context-free parsing strategies were used in our experiments (Manning et al., 1999; Charniak, 1996; Collins, 1999). Extraction of PCFG rules from a treebank is straightforward and we use maximum likelihood estimation to estimate the rule probabilities, as in (Charniak, 1996):

$$\hat{P}(N^i \rightarrow x^j) = \frac{C(N^i \rightarrow x^j)}{\sum_k C(N^i \rightarrow x^k)}$$

Based on the maximum likelihood estimation, we calculate the probabilities of rules in the four levels which are extracted form the 38,944 trees. The PCFG Parser uses these probabilities as its foundation. The standard evaluation and explanation of the parsing result is mentioned in (Manning et al., 1999). The following table shows the result in terms of LP(Labeled Precision), LR(Labeled Recall), LF(Labeled F-measure), BP(Bracket Precision), BR(Bracket Recall), BF(Bracket F-measure), RC(Rule Coverage-rate). Note that a label contains not only syntactic category but also thematic role of a constituent. In addition, the evaluations restricted on the results for valid outputs only are also provided, i.e. without counting the sentences which have no valid parsing results. They are LF-1 and BF-1.

$$LP = \frac{\text{\# correct constituents in parser's parse of S}}{\text{\# constituents in treebank's parse of S}}$$

$$LR = \frac{\text{\# correct constituents in parser's parse of S}}{\text{\# constituents in parser's parse of S}}$$

$$F-measure = \frac{\text{Precision * Recall * 2}}{\text{Precision + Recall}}$$

The parser adopts a top-down Early Algorithm. We modify the representation of the data in order to be applicable in our Sinica Treebank. Two testing data, EV-7 and EV-8, are randomly selected from newly developed Treebank outside of Sinica Treebank Version 2.0. Table 7 and 8 show results of the parsing evaluation respectively.

Ev-7	Case-1	Case-2	Case-3	Case-4
GC	50.59	89.41	93.33	95.98
NP	*	10.33	0.71	0
LR	*	73.03	75.31	59.24
LP	*	71.29	74.53	60.92
LF	*	72.15	74.91	60.07
LF-1	*	80.46	75.45	60.21
BR	*	83.70	91.47	83.80
BP	*	80.48	89.70	85.74
BF	*	82.06	90.58	84.76
BF-1	*	91.51	91.23	84.96

Table 7. EV-7 Sinica Treebank Result (38944 training, 842 testing)

Ev-8	Case-1	Case-2	Case-3	Case-4
GC	48.59	88.26	93.07	95.74
NP	*	9.36	0.44	0
LR	*	71.75	75.79	60.78
LP	*	69.5	74.90	62.16
LF	*	70.73	75.34	61.46
LF-1	*	78.04	75.68	61.46

Ev-8	Case-1	Case-2	Case-3	Case-4
BR	*	83.97	91.83	84.02
BP	*	79.53	89.77	86.42
BF	*	81.69	90.79	85.20
BF-1	*	90.13	91.19	85.20

Table 8. EV-8 Sinica Treebank Result (38944 training, 908 testing)

From Table 7 and 8, we can see that Case-4 has highest grammar coverage (GC), but lowest LF and BF due to higher rule-ambiguities. For the Case-2 model, LF-1 has the best result of 80.46%. However, 10.33% of the sentences are not able to be parsed due to the lower coverage of grammar rules. Case-3 model achieves the best overall performance for its balancing in rule coverage, rule precision and ambiguity. Therefore, the granularity of the rules contributes to the parser accuracy. In general, finer-grained models outperform coarser-grain models, but they also suffer the problem of low grammar coverage. The better parsing performance should be stretched by using more knowledge other than rule probabilities and by considering tradeoffs between grammar coverage and precision.

5 Conclusions and future work

Text annotation is for the purpose of making implicit knowledge in documents more explicit and thus the annotated documents will be easy for processing knowledge extraction. Treebanks provide an easy way of extracting grammar rules and their occurrence probability. In addition, head-modifier and head-argument relations provide the knowledge which is hardly acquired manually. However in our study we also show that for better grammar extraction, a much larger size treebank is required. To construct a very large manually edited treebank is time consuming. We suggest that the knowledge extraction process can be carried out iteratively. The parser can use the coarse-grain grammar and category-to-category relations, which are generalized from word-to-word relations, to produce large amount of automatically parsed trees. The category-to-category relations help to resolve ambiguity of coarse-grain grammar. The newly parsed trees would not produce any new grammar pattern, but they do provide lots of new word-to-word relations. The newly learned relations will increase the knowledge of the parser and hence increase the power of parsing. The whole iteration process can be viewed as a automatic knowledge learning system.

In this study, we also designed a Treebank Searching system. The system provides the users with “Keyword Search” and “Sentence structure search”. Users can further process filtering and aggregating the results within a designated range. By using the Treebank Searching System, we also found some annotation errors in the original treebank. Such information can be discovered from the low frequency syntactic patterns. Therefore, the original treebank is improved after the discovered errors were corrected.

The grammar extraction experiments were carried out. The results indicate that the fine-grain rules have the least coverage rate, while the coarse-grain rules have the higher coverage rate. The coverage rate increases when the size of treebank increases. The fine-grain grammar has less representational ambiguity and the coarse-grain grammar is more ambiguous.

The parsing results reveal that there is plenty of room for exalting the tree bracketing. The relation-knowledge and function word characteristics may help to resolve the some construction ambiguity. We will aim at the individual word and category property and try to increase rule coverage rate by hybrid using Category Level and Simplified Category Level. Our future goal is to improve the parsing rate and maintain the high performance of the parser.

References

E. Charniak. 1996. *Treebank grammars*. Technical Report CS-96-02, Department of Computer Science, Brown University.

Keh-Jiann Chen. 1992. *Design Concepts for Chinese Parsers*. 3rd International Conference on Chinese Information Processing, pp.1-22.

Keh-Jiann Chen, Shing-Huan Liu, Li-ping Chang, and Yeh-Hao Chin. 1994a. *A Practical Tagger for Chinese Corpora*. Proceedings of ROCLING VII, pp.111-126.

Keh-Jiann Chen and Chu-Ren Huang. 1994b. *Features Constraints in Chinese Language Parsing*. Proceedings of ICCPOL '94, pp. 223-228.

Keh-Jiann Chen. 1996a. *A Model for Robust Chinese Parser*. Computational Linguistics and Chinese Language Processing, 1(1):13-204.

Keh-Jiann Chen, Chu-Ren Huang, Li-Ping Chang, Hui-Li Hsu. 1996b. *Sinica Corpus: Design Methodology for Balanced Corpra*. Proceedings of the 11th Pacific Asia Conference on Language, Information, and Computation (PACLIC II), Seoul Korea, pp.167-176.

Feng-Yi Chen, Pi-Fang Tsai, Keh-Jiann Chen, and Chu-Ren Huang. 2000. *Sinica Treebank. [in Chinese]*. Computational Linguistics and Chinese Language Processing, 4(2):87-103.

CKIP (Chinese Knowledge Information Processing). 1993. *The Categorical Analysis of Chinese. [in Chinese]*. CKIP Technical Report 93-05. Nankang: Academia Sinica.

M. Collins. 1999. *Head-Driven Statistical Models for Natural Language parsing*. Ph.D. thesis, Univ. of Pennsylvania.

G. Gazdar, E. Klein, G..K. Pullum, and I. A. Sag. 1985. *Generalized Phrase Structure Grammar*. Cambridge: Blackwell, and Cambridge, Mass: Harvard University Press.

Christopher D. Manning and Hinrich Schutze. 1999. *Foundations of Statistical Natural Language Processing*. the MIT Press, Cambridge, Massachusetts.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. *Building a large annotated corpus of English: The PENN Treebank*. Computational Linguistics, 19(2):313-330.

C. Pollard and I. A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. Stanford: Center for the Study of Language and Information, Chicago Press.

J. Pustejovsky. 1995. *The Generative Lexicon*. MIT Press.

Shin-shyeng Tseng, Meng-yuan Chang, Chin-Chun Hsieh, and Keh-jiann Chen. 1988. *Approaches on An Experimental Chinese Electronic Dictionary*. Proceedings of 1988 International Conference on Computer Processing of Chinese and Oriental Languages, pp. 371-74.

H. Uszkoreit. 1986. *Categorial Unification Grammars*. Proceedings of COLING'86. Bonn: University of Bonn. Also appeared as Report No. CSLI-86-66. Stanford: Center for the Study of Language and Information.

Fei Xia, Martha Palmer, Nianwen Xue, Mary Ellen Okurowski, John Kovarik, Fu-Dong Chiou, Shizhe Huang, Tony Kroch, and Mitch Marcus. 2000. *Developing Guidelines and Ensuring Consistency for Chinese Text Annotation*. Proceedings of the second International Conference on Language Resources and Evaluation (LREC-2000), Athens, Greece.

Yao Yuan and Lua Kim Teng, 1997. *Mutual Information and Trigram Based Merging for Grammar Rule Induction and Sentence parsing*. Computer Processing of Oriental Languages, 11(2):177-190.

Appendix 1. Syntactic Categories

*A ：**NON-PREDICATIVE ADJVECTIVE**

*Caa , Cab, Cba, Cbaa, Cbab, Cbb, Cbba, Cbbb, Cbc, Cbca, Cbcb：**CONJUNCTION**

*Daa, Dab (quantity), Dbaa, Dbab, Dbb, Dbc (modal), Dc (negation), Dd (time), Dfa, Dfb (degree), Dg (locative), Dh (manner), Di (aspect), Dj (interrogative), Dk (sentential adverb)：**ADVERB**

*I ：**INTERJECTION**

Naa (Mass Noun), Nab (Common Noun), Nac(Abstract Noun, Countable), Nad (Abstract Noun), Naea, Naeb(Group Noun), Nba, Nbc(Proper Noun), Nca, Ncb, Ncc, Ncda, Ncdb(Location Noun), Nd(Time Noun) ：**NOUN**

*Neu, Nes, Nep, Neqa, Neqb ：**DETERMINATIVE**

*Nfa, Nfb, Nfc, Nfd, Nfe, Nff, Nfg, Nfh, Nfi ：**MEASURE WORD / CLASSIFIER**

*Ng ：**POSTPOSITION WORD**

*Nhaa, Nhab, Nhac, Nhb, Nhc ：**PRONOUN**

*P01 ~ P65 ：**PREPOSITION**

*Ta, Tb, Tc, Td：**PARTICLE**

[VERB]

*VA11, VA12, VA13, VA2, VA3, VA4：**ACTIVE INTRANSITIVE VERB**

*VB11, VB12, VB2 ： **PSEUDO ACTIVE TRANSITIVE VERB**

*VC1, VC2, VC31, VC32, VC33 ： **ACTIVE TRANSITIVE VERB**

*VD1, VD2：**DITRANSITIVE VERB**

*VE11, VE12, VE2 ：**ACTIVE VERB WITH SENTENTIAL OBJECT**

*VF1, VF2 ： **ACTIVE VERB WITH VP OBJECT**

*VG1, VG2：**CLASSIFICATORY VERB**

*VH11, VH12, VH13, VH14, VH15, VH16, VH17, VH21, VH22 ： **STATIVE INTRANSITIVE**

VERB

*VI1, VI2, VI3 : **PSEUDO STATIVE TRANSITIVE VERB**

*VJ1, VJ2, VJ3 : **STATIVE TRANSITIVE VERB**

*VK1, VK2 : **STATIVE VERB WITH SENTENTIAL OBJECT**

*VL1, VL2, VL3, VL4 : **STATIVE VERB WITH VP OBJECT**

Appendix 2. Syntactic Category Mapping

Level-2	**Level-3**	**Level-4**
Caa	Caa	C
Cab	Cab	C
Cba	Cba	C
Cbaa	Cbb	C
Cbab	Cba	C
Cbba	Cbb	C
Cbbb	Cbb	C
Cbca	Cbb	C
Cbcb	Cbb	C
D*	D	D
Dab	Da	D
DE	DE	DE
Dfa	Dfa	D
Dfb	Dfb	D
Dk	Dk	D
I	I	I
Na*	Na	N
Nb*	Nb	N
Nc*	Nc	N
Ncd*	Ncd	N
Nd*	Nd	N
Nep	Nep	Ne
Neqa	Neqa	Ne
Neqb	Neqb	Ne
Nes	Nes	Ne
Neu	Neu	Ne
Nf*	Nf	N
Ng	Ng	Ng
Nh*	Nh	N
Nv1	Nv	N
Nv2	Nv	N
Nv3	Nv	N
Nv4	Nv	N
P*	P	P
T*	T	T
V_11	SHI	V
V_12	SHI	V
V_2	V_2	V
VA*	VA	V
VA2	VAC	V
VB*	VB	V
VC1	VCL	V
VC*	VC	V
VD*	VD	V
VE*	VE	V
VF*	VF	V
VG*	VG	V
VH*	VH	V
VH16	VHC	V
VH22	VHC	V
VI*	VI	V
VJ*	VJ	V
VK*	VK	V
VL*	VL	V
DM	DM	DM
Di(*)	DE	DE

Appendix 3. ‘jiao’ grammar extraction

From our Tree-searching system, we can find the sentence structures as

VP(Head:VF2:jiao|goal:NP|theme:VP)
S(agent:NP|Head:VG1:jiao|theme:NP|range:NP)
head:VP(Head:VL4:jiao|goal:NP|theme:VP)
VP(Head:VL4:jiao|goal:NP|theme:VP)
S(theme:NP|Head:VG1:jiao|range:NP)
S(agent:NP|Head:VF2:jiao|goal:NP|theme:VP)
complement:VP(Head:VG1:jiao|theme:NP|range:NP)
complement:VP(Head:VG1:jiao|range:NP)

The grammar rule as:

VF2: *<goal[NP]<theme[VP]
agent[NP]<*<goal[NP]<theme[VP]

VG1: agent[NP]<*<theme[NP]<range[NP]
theme[NP]<*<range[NP]
*<theme[NP]<range[NP]
*<range[NP]

VL4: *<goal[NP]<theme[VP]