Linguistically-Motivated Grammar Extraction, Generalization and Adaptation

Yu-Ming Hsieh, Duen-Chi Yang, and Keh-Jiann Chen

Institute of Information Science, Academia Sinica, Taipei
{morris, ydc, kchen}@iis.sinica.edu.tw

Abstract. In order to obtain a high precision and high coverage grammar, we proposed a model to measure grammar coverage and designed a PCFG parser to measure efficiency of the grammar. To generalize grammars, a grammar binarization method was proposed to increase the coverage of a probabilistic context-free grammar. In the mean time linguistically-motivated feature constraints were added into grammar rules to maintain precision of the grammar. The generalized grammar increases grammar coverage from 93% to 99% and bracketing F-score from 87% to 91% in parsing Chinese sentences. To cope with error propagations due to word segmentation and part-of-speech tagging errors, we also proposed a grammar blending method to adapt to such errors. The blended grammar can reduce about 20~30% of parsing errors due to error assignment of pos made by a word segmentation system.

Keywords: Grammar Coverage, Ambiguity, Sentence Parsing, Grammar Extraction.

1 Introduction

Treebanks provide instances of phrasal structures and their statistical distributions. However none of treebanks provide sufficient amount of samples which cover all types of phrasal structures, in particular, for the languages without inflectional markers, such as Chinese. It results that grammars directly extracted from treebanks suffer low coverage rate and low precision [7]. However arbitrarily generalizing applicable rule patterns may cause over-generation and increase ambiguities. It may not improve parsing performance [7]. Therefore a new approach of grammar binarization was proposed in this paper. The binarized grammars were derived from probabilistic context-free grammars (PCFG) by rule binarization. The approach was motivated by the linguistic fact that adjuncts could be arbitrarily occurred or not occurred in a phrase. The binarized grammars have better coverage than the original grammars directly extracted from treebank. However they also suffer problems of over-generation and structure-ambiguity. Contemporary grammar formalisms, such as GPSG, LFG, HPSG, take phrase structure rules as backbone for phrase structure representation and adding feature constraints to eliminate illegal or non-logical structures. In order to achieve higher coverage, the backbone grammar rules (syntactic grammar) are allowed to be over-generation and the feature constraints (semantic grammar for world knowledge) eliminate superfluous structures

R. Dale et al. (Eds.): IJCNLP 2005, LNAI 3651, pp. 177 – 187, 2005.

and increase the precision of grammar representation. Recently, probabilistic preferences for grammar rules were incorporated to resolve structure-ambiguities and had great improvements on parsing performances [2, 6, 10]. Regarding feature constrains, it was shown that contexture information of categories of neighboring nodes, mother nodes, or head words are useful for improving grammar precision and parsing performances [1, 2, 7, 10, 12]. However tradeoffs between grammar coverage and grammar precision are always inevitable. Excessive grammatical constraints will reduce grammar coverage and hence reduce parsing performances. On the other hand, loosely constrained grammars cause structure-ambiguities and also reduce parsing performances. In this paper, we consider grammar optimization in particular for Chinese language. Linguistically-motivated feature constraints were added to the grammar rules and evaluated to maintain both grammar coverage and precision. In section 2, the experimental environments were introduced. Grammar generalization and specialization methods were discussed in section 3. Grammars adapting to pos-tagging errors were discussed in section 4. Conclusions and future researches were stated in the last section.

2 Research Environments

The complete research environment, as shown in the figure 1, comprises of the following five modules and functions.

a) Word segmentation module: identify words including out-of-vocabulary word and provide their syntactic categories.
b) Grammar construction module: extract and derive (perform rule generalization, specialization and adaptation processes) probabilistic grammars from treebanks.
c) PCFG parser: parse input sentences.
d) Evaluation module: evaluate performances of parsers and grammars.
e) Semantic role assignment module: resolve semantic relations for constituents.

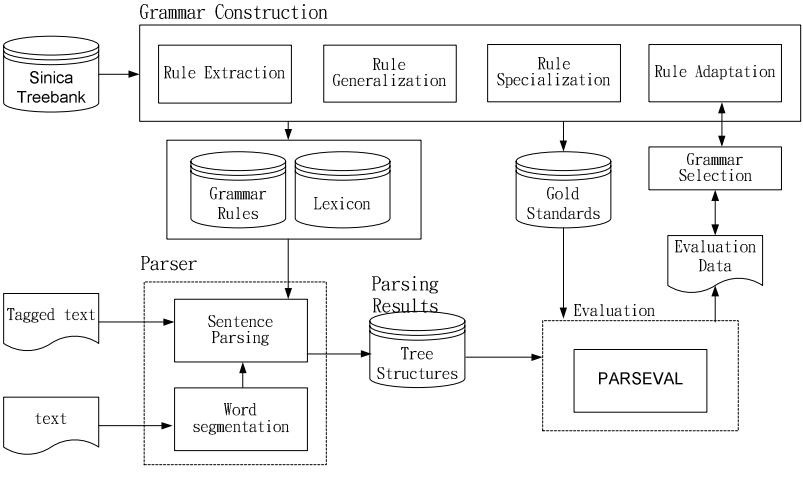

Fig. 1. The system diagram of CKIP parsing environment

2.1 Grammar Extraction Module

Grammars are extracted from Sinica Treebank [4, 5]. Sinica Treebank version 2.0 contains 38,944 tree-structures and 230,979 words. It provides instances of phrasal structures and their statistical distributions. In Sinica Treebank, each sentence is annotated with its syntactic structure and semantic roles for constituents in a dependency framework. Figure 2 is an example.

e.g. 他 叫 李四 撿 球
Ta jiao Li-si jian qiu.
"He asked Lisi to pick up the ball."
Tree-structure:
S(agent:NP(Head:Nh:他)|Head:VF:叫|goal:NP(Head:Nb:李四)|theme:VP(Head:VC:撿|goal:NP(Head:Na:球)))

Fig. 2. A sample tree-structure

Since the Treebank cannot provide sufficient amount of samples which cover all types of phrasal structures, it results that grammars directly extracted from treebanks suffer low coverage rate [5]. Therefore grammar generalization and specialization processes are carried out to obtain grammars with better coverage and precision. The detail processes will be discussed in section 3.

2.2 PCFG Parser and Grammar Performance Evaluation

The probabilistic context-free parsing strategies were used as our parsing model [2, 6, 8]. Calculating probabilities of rules from a treebank is straightforward and we use maximum likelihood estimation to estimate the rule probabilities, as in [2]. The parser adopts an Earley's Algorithm [8]. It is a top-down left-to-right algorithm. The results of binary structures will be normalized into a regular phrase structures by removing intermediate nodes, if used grammars are binarized grammars. Grammar efficiency will be evaluated according to its parsing performance.

2.3 Experiments and Performance Evaluation

Three sets of testing data were used in our performance evaluation. Their basic statistics are shown in Table 1. Each set of testing data represents easy, hard and moderate respectively.

Table 1. Three sets of testing data were used in our experiments

Testing data	Sources	hardness	# of short sentence (1-5 words)	# of normal sentences (6-10 words)	# of long sentences (>11 words)	Total sentences
Sinica	Balanced corpus	moderate	612	385	124	1,121
Sinorama	Magazine	harder	428	424	104	956
Textbook	Elementary school	easy	1,159	566	25	1,750

The following parser and grammar performance evaluation indicators were used in our experiments:

- LP(Labeled Precision)
$$LP = \frac{\#\text{ of correct phrases labeled by the parser}}{\#\text{ of phrases labeled by the parser}}$$
- LR(Labeled Recall)
$$LR = \frac{\#\text{ of correct phrases labeled by the parser}}{\#\text{ of phrases in the testing data}}$$
- LF(Labeled F-measure)
$$LF = \frac{LP * LR * 2}{LP + LR}$$
- BP(Bracketed Precision)
$$BP = \frac{\#\text{ of pairs of brackets correctly made by the parser}}{\#\text{ of pairs of brackets made by the parser}}$$
- BR(Bracketed Recall)
$$BR = \frac{\#\text{ of pairs of brackets correctly made by the parser}}{\#\text{ of pairs of brackets in the gold standard of the testing data}}$$
- BF(Bracketed F-measure)
$$BF = \frac{BP * BR * 2}{BP + BR}$$

Additional indicators regarding coverage of grammars:

- RC-Type：type coverage of rules
$$RC\text{-}Type = \frac{\#\text{ of rules types in both testing data and grammar rules}}{\#\text{ of rule types in testing data}}$$
- RC-Token：token coverage of rules
$$RC\text{-}Token = \frac{\#\text{ of rules tokens in both testing data and grammar rules}}{\#\text{ of rule tokens in testing data}}$$

The token coverage of a set of rules is the ceiling of parsing algorithm to achieve. Tradeoff effects between grammar coverage and parsing F-score can be examined for each set of rules.

3 Grammar Generalization and Specialization

By using above mentioned research environment, we intend to find out most effective grammar generalization method and specialization features for Chinese language. To extend an existing or extracted grammar, there are several different approaches. A naïve approach is to generalize a fine-grained rule to a coarse-grained rule. The approach does not generate new patterns. Only the applicable patterns for each word were increased. However it was shown that arbitrarily increasing the applicable rule patterns does increase the coverage rates of grammars, but degrade parsing performance [5]. A better approach is to generalizing and specializing rules under linguistically-motivated way.

3.1 Binary Grammar Generation, Generalization, and Specialization

The length of a phrase in Treebank is variable and usually long phrases suffer from low probability. Therefore most PCFG approaches adopt the binary equivalence grammar, such as Chomsky normal form (CNF). For instance, a grammar rule of S→ NP Pp Adv V can be replaced by the set of equivalent rules of {S→Np R0, R0→Pp R1, R1→Adv V}. The binarization method proposed in our system is different from CNF. It generalizes the original grammar to broader coverage. For instance, the above rule after performing right-association binarization[1] will produce following three binary rules {S→Np S', S'→Pp S', S'→Adv V}. It results that constituents (adjuncts and arguments) can be occurred or not occurred at almost any place in the phrase. It partially fulfilled the linguistic fact that adjuncts in a phrase are arbitrarily occurred. However it also violated the fact that arguments do not arbitrarily occur. Experimental results of the Sinica testing data showed that the grammar token coverage increased from 92.8% to 99.4%, but the labeling F-score dropped from 82.43% to 82.11% [7]. Therefore feature constraints were added into binary rules to limit over-generation caused by recursively adding constituents into intermediate-phrase types, such as S' at above example.

Feature attached rules will look like following:

$S'_{\text{-left:Adv-head:V}} \rightarrow$ Adv V;
$S'_{\text{-left:Pp-head:V}} \rightarrow$ Pp $S'_{\text{-left:Adv-head:V}}$;

The intermediated node $S'_{\text{-left:Pp-head:V}}$ says that it is a partial S structure with left-most constituent Pp and a phrasal head V. Here the leftmost feature constraints linear order of constituents and the head feature implies that the structure patterns are head word dependent. Both constraints are linguistically plausible. Another advantage of the feature-constraint binary grammar is that in addition to rule probability it is easy to implement association strength of modifier word and head word to evaluate plausibility of derived structures.

3.2 Feature Constraints for Reducing Ambiguities of Generalized Grammars

Adding feature constraints into grammar rules attempts to increase precision of grammar representation. However the side-effect is that it also reduces grammar coverage. Therefore grammar design is balanced between its precision and coverage. We are looking for a grammar with highest coverage and precision. The tradeoff depends on the ambiguity resolution power of adopted parser. If the ambiguity resolution power of adopted parser is strong and robust, the grammar coverage might be more important than grammar precision. On the other hand a weak parser had better to use grammars with more feature constraints. In our experiments, we consider grammars suited for PCFG parsing. The follows are some of the most important linguistically-motivated features which have been tested.

[1] The reason for using right-association binarization instead of left-association or head-first association binarization is that our parsing process is from left to right. It turns out that parsing speed of right associated grammars is much faster than left-associated grammars for left-to-right parsing.

Head (*Head feature*): Pos of phrasal head will propagate to all intermediate nodes within the constituent.
Example:S(NP(Head:Nh:他)|S'$_{-VF}$(Head:VF:叫|S'$_{-VF}$(NP(Head:Nb:李四)|
VP(Head:VC:撿| NP(Head:Na:球)))))
Linguistic motivations: Constrain sub-categorization frame.

Left (*Leftmost feature*): The pos of the leftmost constitute will propagate one–level to its intermediate mother-node only.
Example:S(NP(Head:Nh:他)|S'$_{-Head:VF}$(Head:VF:叫|S'$_{-NP}$(NP(Head:Nb:李四)|
VP(Head:VC:撿| NP(Head:Na:球)))))
Linguistic motivation: Constraint linear order of constituents.

Mother (*Mother-node*): The pos of mother-node assigns to all daughter nodes.
Example:S(NP$_{-S}$(Head:Nh:他)|S'(Head:VF:叫|S'(NP$_{-S}$(Head:Nb:李四)|VP$_{-S}$(Head:VC:撿| NP$_{-VP}$(Head:Na: 球)))))
Linguistic motivation: Constraint syntactic structures for daughter nodes.

Head0/1 (*Existence of phrasal head*): If phrasal head exists in intermediate node, the nodes will be marked with feature 1; otherwise 0.
Example:S(NP(Head:Nh: 他)|S'$_{-1}$(Head:VF: 叫 |S'$_{-0}$(NP(Head:Nb: 李 四)|VP(Head:VC: 撿| NP(Head:Na: 球)))))
Linguistic motivation: Enforce unique phrasal head in each phrase.

Table 2. Performance evaluations for different features

	(a)Binary rules without features			(b)Binary+Left		
	Sinica	Snorama	Textbook	Sinica	Sinorama	Textbook
RC-Type	95.632	94.026	94.479	95.074	93.823	94.464
RC-Token	99.422	99.139	99.417	99.012	98.756	99.179
LP	81.51	77.45	84.42	86.27	80.28	86.67
LR	82.73	77.03	85.09	86.18	80.00	87.23
LF	82.11	77.24	84.75	86.22	80.14	86.94
BP	87.73	85.31	89.66	90.43	86.71	90.84
BR	89.16	84.91	90.52	90.46	86.41	91.57
BF	88.44	85.11	90.09	90.45	86.56	91.20
	(c)Binary+Head			(d)Binary+Mother		
	Sinica	Snorama	Textbook	Sinica	Sinorama	Textbook
RC-Type	94.595	93.474	94.480	94.737	94.082	92.985
RC-Token	98.919	98.740	99.215	98.919	98.628	98.857
LP	83.68	77.96	85.52	81.87	78.00	83.77
LR	83.75	77.83	86.10	82.83	76.95	84.58
LF	83.71	77.90	85.81	82.35	77.47	84.17
BP	89.49	85.29	90.17	87.85	85.44	88.47
BR	89.59	85.15	90.91	88.84	84.66	89.57
BF	89.54	85.22	90.54	88.34	85.05	89.01

Each set of feature constraint added grammar is tested and evaluated. Table 2 shows the experimental results. Since all features have their own linguistic motivations, the result feature constrained grammars maintain high coverage and have improving grammar precision. Therefore each feature more or less improves the parsing performance and the feature of leftmost daughter node, which constrains the linear order of constituents, is the most effective feature. The Left-constraint-added grammar reduces grammar token-coverage very little and significantly increases label and bracket f-scores.

It is shown that all linguistically-motivated features are more or less effective. The leftmost constitute feature, which constraints linear order of constituents, is the most effective feature. The mother-node feature is the least effective feature, since syntactic structures do not vary too much for each phrase type while playing different grammatical functions in Chinese.

Table 3. Performances of grammars with different feature combinations

	(a) Binary+Left+Head1/0			(b) Binary+Left+Head		
	Sinica	Sinorama	Textbook	Sinica	Sinorama	Textbook
RC-Type	94.887	93.745	94.381	92.879	91.853	92.324
RC-Token	98.975	98.740	99.167	98.173	98.022	98.608
LF	86.54	79.81	87.68	86.00	79.53	86.86
BF	90.69	86.16	91.39	90.10	86.06	90.91
LF-1	86.71	79.98	87.73	86.76	79.86	87.16
BF-1	90.86	86.34	91.45	90.89	86.42	91.22

Table 4. Performances of the grammar with most feature constraints

	Binary+Left+Head+Mother+Head1/0		
	Sinica	Sinorama	Textbook
RC-Type	90.709	90.460	90.538
RC-Token	96.906	96.698	97.643
LF	86.75	78.38	86.19
BF	90.54	85.20	90.07
LF-1	88.56	79.55	87.84
BF-1	92.44	86.46	91.80

Since all the above features are effective, we like to see the results of multi-feature combinations. Many different feature combinations were tested. The experimental results show that none of the feature combinations outperform the binary grammars with Left and Head1/0 features, even the grammar combining all features, as shown in the Table 3 and 4. Here LF-1 and BF-1 measure the label and bracket f-scores only on the sentences with parsing results (i.e. sentences failed of producing parsing results are ignored). The results show that grammar with all feature constraints has better LF-1 and BF-1 scores, since the grammar has higher precision. However the total performances, i.e. Lf and BF scores, are not better than the simpler grammar with feature

constraints of Left and Head1/0, since the higher precision grammar losses slight edge on the grammar coverage. The result clearly shows that tradeoffs do exist between grammar precision and coverage. It also suggests that if a feature constraint can improve grammar precision a lot but also reduce grammar coverage a lot, it is better to treat such feature constraints as a soft constraint instead of hard constraint. Probabilistic preference for such feature parameters will be a possible implementation of soft constraint.

3.3 Discussions

Feature constraints impose additional constraints between constituents for phrase structures. However different feature constraints serve for different functions and have different feature assignment principles. Some features serve for local constraints, such as Left, Head, and Head0/1. Those features are only assigned at local intermediate nodes. Some features are designed for external effect such as Mother Feature, which is assigned to phrase nodes and their daughter intermediate nodes. For instances, NP structures for subject usually are different from NP structures for object in English sentences [10]. NP attached with Mother-feature can make the difference. NP_S rules and NP_{VP} rules will be derived each respectively from subject NP and object NP structures. However such difference seems not very significant in Chinese. Therefore feature selection and assignment should be linguistically-motivated as shown in our experiments.

In conclusion, linguistically-motivated features have better effects on parsing performances than arbitrarily selected features, since they increase grammar precision, but only reduce grammar coverage slightly. The feature of leftmost daughter, which constraints linear order of constituents, is the most effective feature for parsing. Other sub-categorization related features, such as mother node and head features, do not contribute parsing F-scores very much. Such features might be useful for purpose of sentence generation instead of parsing.

4 Adapt to Pos Errors Due to Automatic Pos Tagging

Perfect testing data was used for the above experiments without considering word segmentation and pos tagging errors. However in real life word segmentation and pos tagging errors will degenerate parsing performances. The real parsing performances of accepting input from automatic word segmentation and pos tagging system are shown in the Table 5.

Table 5. Parsing performances of inputs produced by the automatic word segmentation and pos tagging

	Binary+Left+Head1/0		
	Sinica	Sinorama	Textbook
LF	76.18	64.53	73.61
BF	84.01	75.95	84.28

The naïve approach to overcome the pos tagging errors was to delay some of the ambiguous pos resolution for words with lower confidence tagging scores and leave parser to resolve the ambiguous pos until parsing stage. The tagging confidence of each word is measured by the following value.

Confidence value= $\frac{P(c_{1,w})}{P(c_{1,w})+P(c_{2,w})}$, where $P(c_{1,w})$ and $P(c_{2,w})$ are probabilities assigned by the tagging model for the best candidate $c_{1,w}$ and the second best candidate $c_{2,w}$.

The experimental results, Table 6, show that delaying ambiguous pos resolution does not improve parsing performances, since pos ambiguities increase structure ambiguities and the parser is not robust enough to select the best tagging sequence. The higher confidence values mean that more words with lower confidence tagging will leave ambiguous pos tags and the results show the worse performances. Charniak et al [3] experimented with using multiple tags per word as input to a treebank parser, and came to a similar conclusion.

Table 6. Parsing performances for different confidence level of pos ambiguities

	Confidence value=0.5		
	Sinica	Sinorama	Textbook
LF	75.92	64.14	74.66
BF	83.48	75.22	83.65
	Confidence value=0.8		
	Sinica	Sinorama	Textbook
LF	75.37	63.17	73.76
BF	83.32	74.50	83.33
	Confidence value=1.0		
	Sinica	Sinorama	Textbook
LF	74.12	61.25	69.44
BF	82.57	73.17	81.17

4.1 Blending Grammars

A new approach of grammar blending method was proposed to cope with pos tagging errors. The idea is to blend the original grammar with a newly extracted grammar derived from the Treebank in which pos categories are tagged by the automatic pos tagger. The blended grammars contain the original rules and the extended rules due to pos tagging errors. A 5-fold cross-validation was applied on the testing data to tune the blending weight between the original grammar and the error-adapted grammar. The experimental results show that the blended grammar of weights 8:2 between the original grammar and error-adapted grammar achieves the best results. It reduces about 20%~30% parsing errors due to pos tagging errors, shown in the Table 7. The pure error-adapted grammar, i.e. 0:10 blending weight, does not improve the parsing performance very much

Table 7. Performances of the blended grammars

	Error-adapted grammar i.e. blending weight (0:10)			Blending weight 8:2		
	Sinica	Sinirama	Textbook	Sinica	Sinirama	Textbook
LF	75.99	66.16	71.92	78.04	66.49	74.69
BF	85.65	77.89	85.04	86.06	77.82	85.91

5 Conclusion and Future Researches

In order to obtain a high precision and high coverage grammar, we proposed a model to measure grammar coverage and designed a PCFG parser to measure efficiency of the grammar. Grammar binarization method was proposed to generalize rules and to increase the coverage of context-free grammars. Linguistically-motivated feature constraints were added into grammar rules to maintain grammar rule precision. It is shown that the feature of leftmost daughter, which constraints linear order of constituents, is the most effective feature. Other sub-categorization related features, such as mother node and head features, do not contribute parsing F-scores very much. Such features might be very useful for purpose of sentence generation instead of parsing. The best performed feature constraint binarized grammar increases the grammar coverage of the original grammar from 93% to 99% and bracketing F-score from 87% to 91% in parsing moderate hard testing data. To cope with error propagations due to word segmentation and part-of-speech tagging errors, a grammar blending method was proposed to adapt to such errors. The blended grammar can reduce about 20~30% of parsing errors due to error assignment of a pos tagging system.

In the future, we will study more effective way to resolve structure ambiguities. In particular, consider the tradeoff effect between grammar coverage and precision. The balance between soft constraints and hard constraints will be focus of our future researches. In addition to rule probability, word association probability will be another preference measure to resolve structure ambiguity, in particular for conjunctive structures.

Acknowledgement

This research was supported in part by National Science Council under a Center Excellence Grant NSC 93-2752-E-001-001-PAE and National Digital Archives Program Grant NSC93-2422-H-001-0004.

References

1. E. Charniak, and G. Carroll, “Context-sensitive statistics for improved grammatical language models.” In Proceedings of the 12th National Conference on Artificial Intelligence, AAAI Press, pp. 742-747, Seattle, WA, 1994,
2. E. Charniak, “Treebank grammars.” In Proceedings of the Thirteenth National Conference on Artificial Intelligence, pp. 1031-1036. AAAI Press/MIT Press, 1996.

3. E. Charniak, and G. Carroll, J. Adcock, A. Cassanda, Y. Gotoh, J. Katz, M. Littman, J. Mccann, "Taggers for Parsers", Artificial Intelligence, vol. 85, num. 1-2, 1996.
4. Feng-Yi Chen, Pi-Fang Tsai, Keh-Jiann Chen, and Huang, Chu-Ren, "Sinica Treebank." Computational Linguistics and Chinese Language Processing, 4(2):87-103, 2000.
5. Keh-Jiann Chen and, Yu-Ming Hsieh, "Chinese Treebanks and Grammar Extraction." the First International Joint Conference on Natural Language Processing (IJCNLP-04), March 2004.
6. Michael Collins, "Head-Driven Statistical Models for Natural Language parsing." Ph.D. thesis, Univ. of Pennsylvania, 1999.
7. Yu-Ming Hsieh, Duen-Chi Yang and Keh-Jiann Chen, "Grammar extraction, generalization and specialization. (in Chinese)"Proceedings of ROCLING 2004.
8. Christopher D. Manning and Hinrich Schutze, "Foundations of Statistical Natural Language Processing." the MIT Press, Cambridge, Massachusetts, 1999.
9. Mark Johnson, "PCFG models of linguistic tree representations." Computational Linguistics, Vol.24, pp.613-632, 1998.
10. Dan Klein and Christopher D. Manning, "Accurate Unlexicalized Parsing." Proceeding of the 4lst Annual Meeting of the Association for Computational Linguistics, pp. 423-430, July 2003.
11. Honglin Sun and Daniel Jurafsky, "Shallow Semantic Parsing of Chinese." Proceedings of NAACL 2004.
12. 12.Hao Zhang, Qun Liu, Kevin Zhang, Gang Zou and Shuo Bai, "Statistical Chinese Parser ICTPROP." Technology Report, Institute of Computing Technology, 2003.