

applying the chunker to its own output. Kudo and Yamada (2002) proposed a statistical Japanese dependency parser using a cascaded chunking model. Yoshimasa Tsuruoka et al. (2005) improve the performance of the chunk parsing approach by using a simple sliding-window method and maximum entropy classifiers for phrase recognition in each level of chunking.

Noun phrase recognition plays an important role during the understanding process of the sentence. Entities and concepts are generally described by the noun phrases in texts. Therefore, if we identify the noun phrases of texts, the main meaning of the texts can be easily grasped. In Chinese, a syntactic constituent can modify an NP without any morphological changes, if they are semantically matched, merely requiring a particle “的(de)” added between them. From syntactic function, MNP generally appears in the subject and the object of a sentence. If all MNPs in a sentence are identified, the sentence’s structural frame can be easily obtained and the parsing tree of a sentence can be easily constructed (Qiang Zhou, 2000). Therefore, the identifying and parsing of MNP will be of great help to syntactic parsing.

According to the characteristics of Chinese, this paper proposes a statistical parsing method based on maximal noun phrase per-processing. Maximal noun phrase parsing is separated from parsing. Maximal noun phrases in a sentence are firstly identified, and then the sentence is parsed with the head of the Maximal noun phrases. Therefore, an original sentence is divided into two parts to parse separately. The first part is Maximal noun phrase parsing; the second part is parsing of the sentence in which the Maximal Noun phrases are replaced by their head words. Finally, the paper takes Conditional Random Fields as the statistical recognition model of each level in syntactic parsing process.

The paper is organized as follows: in section 2 we introduce the parsing approach based on Cascaded Conditional Random Fields; in section 3 we describe a statistical parsing method based on maximal noun phrase per-processing; the method of head tagging is briefly introduced in section 4; section 5 analyses the results of experiments; finally, we conclude our work in section 6.

2 Parsing Based on Cascaded Conditional Random Fields

The syntactic parsing process can be converted into many levels, Chinese chunking at the bottom level and phrases identification at the other higher levels. The identified task in each level can be regarded as the sequence tagging problem similar to the Part-of-Speech tagging. Therefore, the syntactic parsing is converted into a multi-level tagging task:

In this paper, Chinese chunks are defined as the smallest syntactic functional units of a sentence, which are grammatical non-recursive phrases and phrases are defined as the constituents of a sentence with nested structure.

2.1 Conditional Random Fields

Conditional Random Fields (CRF), a statistical sequence labeling model, was first introduced by Lafferty et al. (2001). CRF models are undirected graphical models, which calculate the conditional probability of label sequences given input sequences. Unlike generative models, CRF models don’t require stringent conditional independence assumptions because the majority of sequence data can not be represented as a series of independent events. Comparing with other discriminative models like maximum entropy model, CRF models overcome the label bias problem and can trade off decisions at different sequence positions to obtain a globally optimal labeling. CRF has already been applied to tasks such as Part-of-Speech tagging, shallow parsing and named-entity recognition.

The random variable sequence $x = x_1 \cdots x_N$ is defined as an observation value sequence, such as the input Chinese word sequence. The sequence $y = y_1 \cdots y_N$ is defined as an output state sequence, such as an output tagging sequence. The conditional probability of a label sequence y given an input sequence defined by CRF is:

$$P(y|x) = \frac{1}{Z(x)} \exp\left(\sum_{i=1}^N \sum_k \lambda_k f_k(y_{i-1}, y_i, x, t)\right)$$

Where $Z(x)$, a normalization constant, makes the sum of all possible state sequences equal 1. $f_k(y_{i-1}, y_i, x, t)$ is a binary feature function of CRF models. λ_k , the weight parameter correlated with f_k , can be obtained through training, which

indicates how important feature f_k is for the model.

Since CRF as a conditional model can comprehensively use multi-level resources such as character, word and Part-of-Speech and has very good description ability for the long distance connection, this paper takes CRF as the statistical recognition model of each level in syntactic parsing process.

2.2 Cascaded Conditional Random Fields

We can see from the example “dj[np[远古/t 人类/n] vp[vp[生活/v 在/p] sp[np[恶劣/a 的/uJDE 自然环境/n] 中/f]]” that a sentence contains a base noun phrase “np[远古/t 人类/n]” and a verb phrase “vp[vp[生活/v 在/p] sp[np[恶劣/a 的/uJDE 自然环境/n] 中/f]” , and the verb phrase contains a base verb phrase “vp[生活/v 在/p]” and a space phrase “sp[np[恶劣/a 的/uJDE 自然环境/n] 中/f]”, and the space phrase contains a base noun phrase “np[恶劣/a 的/uJDE 自然环境/n]”. Due to the nested structure of the sentence, direct parsing may result in the ambiguity of structural analysis, because words contained in chunks may be combined with its context into phrases. Therefore, we need to build a multi-layer model for parsing. There are two kinds of methods to build this model. One takes the lower model as the higher model’s sub-model, and the other takes the output of the bottom model as the input

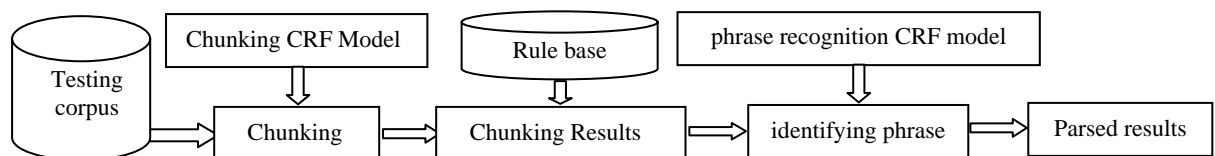


Figure 1. System flowchart of Chinese parsing based on CCRFs

The whole procedure includes three modules: (1) chunk recognition module based on the lower CRF model; (2) rule-based post-processing module; (3) phrase tagging based on the high-level CRF model. First Chinese chunks are identified based on the lower CRF model given a sentence with gold segmentation and Part-of-Speech tagging, and replaced by their types. The chunking result is passed as input to the next phrase recognition level. Because some recognition errors will be introduced at chunking stage, the errors are corrected by post-processing rules constructed before entering the next phrase recognition stage, which aims at reducing the

of the higher model. The former method is used for building a hierarchical model (M. Skounakis, 2003), the latter method which Cascaded Conditional Random Fields models belong to is selected to build a cascaded models (Brants, T., 1999).

Cascaded Conditional Random Fields models are two-stage models constructed by two CRF models. This paper divides the parsing task into many levels and takes Cascaded Conditional Random Fields (CCRFs) model as the parsing model, where CRF is the statistical model of tagging in each level. Chunks are identified based on the lower CRF model given an observation value sequence. Chunking result is passed as input to the high-level CRF model. For the high-level model, input variables contain not only observation values but also chunking results of the lower level.

The training corpus used for training every CRF model are all came from this evaluation training corpus, but we need to change the format of Treebank corpus for training CRF model in each level. The paper utilizes two CRF models: chunking CRF model and phrase recognition model.

2.3 Automatic parsing based on CCRFs

The input for parsing is Chinese character sequence with gold segmentation and Part-of-Speech tagging and the output is the most likely sentence parsing tree.

errors transmitted to the next stage. Phrase recognition is performed based on the high-level CRF model with the result of the previous step as input. The Phrase recognition results are passed as input to the next level until no new phrases can be discovered, and finally the syntactic parsing process completes with the output of sentence parsing tree.

2.4 Feature Selection

The selection of features often plays a crucial role in chunk recognition based on CRF. CRF model can use simple features to represent complex linguistic phenomena without any

independence assumptions. We use lexical, POS information and the different combination of them as the features of chunk tagging according to different factors affecting chunk tagging. In this paper, we set the context window as 2. Feature spaces are listed as follows: (1) POS information, the POS of previous two words, current word and next two words. (2) Word information, previous two words, current word and next two words.

According to these features, we define the template, which consists of atom templates and complex templates. The atom template can be seen as a feature function of the context.

Number	Atom Template	Definition
1	CurWord	Current word
2	CurPOSTag	POS of current word
3	Word-2	Context of the current word
4	Word-1	
5	Word+1	
6	Word+2	
7	POSTag-2	POS of context of the current word
8	POSTag-1	
9	POSTag+1	
10	POSTag+2	

Table 1. Atom Template

It is not enough to identify some of the phenomena of the context by using only atom features. We can describe more complex context by some complex templates which are combined by the atom features in the above list. The atom templates in the complex features templates can be instantiated when the feature function takes a specific value, which gets specific features. The complex templates of the chunking in the lower level have five kinds. The complex templates of the phrase recognition in the higher level have twelve kinds.

All the feature templates of models are composed of the atom feature templates and the complex templates. The chunking in the lower level has 15 template types, and the phrase recognition in the higher level has 22 template types.

3 Statistical parsing based on Maximal Noun Phrase pre-processing

The parsing process consists of three statistical models: the MNP recognition model, the chunking model and the phrase recognition model.

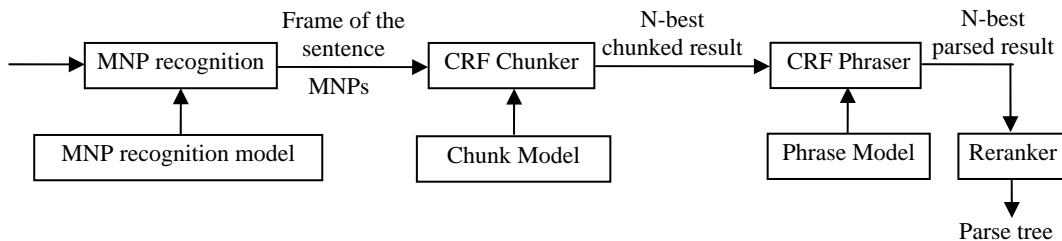


Figure 2. System architecture of Parser based on Maximal Noun Phrase pre-processing

3.1 MNP Recognition

Maximal noun phrase (MNP) is the noun phrase which is not contained by any other noun phrases. The paper uses the method of statistic plus rules to identify MNP. First, the CRF model are built according to feature selection and parameter estimation on the basis of training corpus, MNPs in the unlabeled corpus are identified based on CRF model, and the recognition result are

obtained primarily; second, we process the recognition result using post-processing rules and obtain the final recognition result. The MNP recognition method in the paper primarily come from the reference (Cui Dai, 2008), but the feature templates are slightly different, for which we will explain later in this paper, and details of the other parts can be found in reference (Cui Dai, 2008).

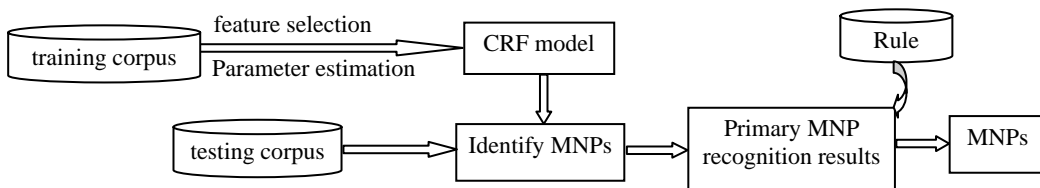


Figure 3. Flow chart of the MNP recognition

We use the lexical, part-of-speech information and the different combinations of them as the feature of the MNP recognition, and we set the context window of templates as 3. The atom feature templates of the MNP recognition contain 14 types: CurWord, CurPOSTag, Word-3, Word-2, Word-1, Word+1, Word+2, Word+3, POSTag-3, POSTag-2, POSTag-1, POSTag+1, POSTag+2, POSTag+3. The complex feature templates contain 13 types: Word-1/CurWord, POSTag-1/CurPOSTag, CurWord/Word+1, CurPOSTag/POSTag+1, Word-2/POSTag-2, CurWord/CurPOSTag, Word+3/POSTag+3, POSTag-2/POSTag-1/CurPOSTag, POSTag-1/CurPOSTag/POSTag+1,

POSTag+1/POSTag+2/POSTag+3, POSTag-3/POSTag-2, POSTag-2/POSTag-1, Word+2/Word+3. The atom feature templates and the complex feature templates together continue the feature templates of the MNP recognition with 27 template types.

The MNP training corpus and test corpus are provided by the evaluation. We classify MNPs into three classes in our experiments: Simple MNP (SMNP) whose length is less than five, and Complex MNP (CMNP) whose length is no less than five, CMNPs are classified into two classes: one class whose length is less than 10 and the other one whose length is no less than 10.

Corpus	Average Length	SMNP	CMNP(5≤len<10)	CMNP(len≥10)
training corpus	5.58	21585	12710	4801
testing corpus	5.33	5577	3097	1055

Table 2. Distribution of experimental corpus

Method	N1	N2	N3	P	R	F
CRFs+rules	9729	9669	8493	87.84%	87.3%	87.57%
CRFs	9729	9580	8081	84.35%	83.06%	83.7%

Table 3. Comparative results of MNP recognition

We can see from the table 3 that the average length of MNP in the testing corpus is more than 5, thus MNP is more complex in general. F-score of CRFs plus rules method is 5% higher than the single CRFs model method.

3.2 The generation of sentence's frame

In this paper, the new sentence in which MNPs are replaced by their head is defined as the sentence's frame. The heads of MNPs are identified after MNP recognition and then they are used to replace the original MNP, and finally the sentence's frame is formed.

We use the rules to recognize the head of MNP. The last word of MNP is the head of the phrase, which can represent the MNP in function. For example: “[该学派] 同样主张消除[干预造成的阻碍]。” In this sentence “该学派” and “干预造成的阻碍” are MNPs. If we omit the modified components in MNP, for example “[学派]同样主张消除[阻碍]。”, the meaning of the sentence will not be changed. Because the head can represent the syntax function of MNP, we can use the head for parsing, which can avoid the effect of the modifier of MNP on parsing and reduce the complexity of parsing. For example: Original sentence: [dj [np [np 凯恩斯主义 及其 [np 战后 [np 各流派]]]] 的主张] [vp [pp 在

[sp 事实 面前]] [vp 遇到 [np 严重 挫折]]]]
Frame:[dj 主张 [vp [pp 在 [sp 事实 面前]] [vp 遇到 挫折]]]

However, the components of MNP are complicated, not all of the last word of MNP can be the head of MNP. The paper shows that if MNP has parentheses, we can use the last word before parentheses as the head in the following example:

Original sentence: [dj 药物 [vp [pp 以 [np 天然药 [dlc ([vp 包括 [np [np 植物、动物和矿物] 的 [np 药用部分]])]]] 为主]]]

MNP: [np 天然药 (包括植物、动物和矿物的药用部分)]

Frame: [dj 药物 [vp [pp 以 天然药] 为主]]

3.3 Generating parsing tree by Local Optimization Method

We use CCRFs for parsing the MNP and the sentence's frame, and then combine the parsing result of the two parts, and finally form the parsing tree of original sentence. The paper proposes a local optimal search algorithm to find the local optimal solution in the process of generating parsing tree. The lower CRF model is used to identify chunks with the lexical and POS information of words. The N-best chunked results are transferred to the upper CRF models

as input. The upper CRF models are used to perform phrase recognition. Every chunked result is produced 1-best phrase recognition result. The N-best phrase recognition results are passed as input to the next level until no new phrases can be discovered.

$$T^* = \arg \max_{\substack{1 \leq i \leq 5 \\ T_{1i}^* \in T_1^*}} p(T_{1i}^* / W_1, P_1) + \arg \max_{\substack{1 \leq i \leq 5 \\ T_{2i}^* \in T_2^*}} p(T_{2i}^* / W_2, P_2) \quad (1)$$

$$\text{Wherein, } p(T_{ij}^* / W_i, P_i) = \left(\prod_{k=1}^{\text{level of } T_{ij}^*} \log \text{Prob}_k \right) / (\text{level of } T_{ij}^*), i=1,2, 1 \leq j \leq 5 \quad (2)$$

W is the context sequence of the sentence, P is the POS sequence of the sentence, T_1^* and T_2^* are separately the 5-best results of frame parsing and the 5-best results of the MNP parsing. T^* is the best parsing result.

4 Labeling of core syntactic component

The labeling of core syntactic component is to identify the core words of each syntactic component from a sentence and perform labeling. The method of core component labeling used in this paper is independent of the process of parsing. The labeling of core component is performed on the analysis result of parsing tree. The examples are as follows:

Input: [dj 知识/n [vp 就/d [vp 是/vC 力量/n]]]
Output: [dj-1 知识/n [vp-1 就/d [vp-0 是/vC 力量/n]]]

The paper uses method combining statistics and rules to label core component. The selected statistical method is conditional random field model. First, the system trains CRF model on the basis of train corpus by the feature selection and parameter estimation. Core component labeling is performed on unlabeled corpus based on CRFs, and gets the preliminary identification results; then a rule-based method is used to post process, the identification results and gets the final recognition results. The rule-based post-processing module mainly uses rule-base and case-base to carry out post-processing.

4.1 Core component labeling based on CRF

The core components labeling of a sentence is divided into two parts in this paper. One is the core component labeling of underlying chunks, and the other is the core component labeling of upper phrases. Therefore, two models need to be trained, namely the underlying core component

We use the formula (2) for reordering the last N-best parsed results to obtain the optimum parsing result. The local search formulas for parsing are as follows:

labeling model and the upper core component labeling model. In the paper, three features of word, part of speech and phrase type are used to constitute atomic templates and complex templates. The best templates are selected through experiments for automatic head recognition.

4.2 Rule-based core components labeling

Through the analysis of error examples, we found that some CRF recognition results are clearly inconsistent with the actual situation, we can use rules to correct these errors, thus forming a rule base. The main types of errors are as follows:

- (1) [np-0-1-2-3 诗/n 词/n 歌/n 赋/n] [np-0-1-2-3 晋/nS 冀/nS 鲁/nS 豫/nS]
- (2) [np-0-2 荀子/nP ·/w 议兵篇/nR] [np-0-2 孙子/nP ·/w 地形篇/n]
- (3) [vp-2 走/v 一/d 走/v] [vp-2 抓/v 一/m 抓/v] [vp-2 看/v 了/uA 看/v]

We set up rules according to the above error types to perform additional identification and correction, thus increasing the recognition accuracy.

Example-base is a phrase-based library built through analysis and processing on the training corpus. The Example-base is composed of all the bottom phrases and high-level phrases in training corpus. High-level phrases are the bottom phrases replaced by heads, with 168655 phrases. The building process of Example-base is as follows:

[dj-1 [np-1 温病/n 学说/n] [vp-0 达到/v [tp-1 成熟/a 阶段/nT]]]

First, the bottom phrases “[np-1 温病/n 学说/n]” and “[tp-1 成熟/a 阶段/nT]” are added to the case-base, and then the bottom phrases are replaced with heads to generate new bottom phrase “[vp-0 达到/v 阶段/nT]” to added to the

case-base, and so on. Finally we can get a phrase-base library. In this sentence, four phrases are generated: “[np-1 温病/n 学说/n]”, “[tp-1 成熟/a 阶段/nT]”, “[vp-0 达到/v 阶段/nT]”, “[dj-1 学说/n 达到/v]”. Each sentence uses the same approach, and a phrase-base library is built.

4.3 Experiment results of core component labeling

The experimental results of core component labeling on the basis of all correct labeling of the syntax tree component are as follows:

Method	Correct Number	Wrong Number	Total Number	Precision
CRF	61503	826	62329	98.6748%
CRF+rule-base	61518	811	62329	98.6988%
CRF+rule-base+case-base	61556	773	62329	98.7598%

Table 4. Comparative experimental result of core component labeling

Since we only used rules to correct a few simple errors, and the number of rule numbers is limited, the accuracy rate only increased 0.024% by the amended rules. But the accuracy rate increased nearly 0.06% after adding the case-base, which indicates that the case-base method is effective. However, we also found some inconsistencies between the training corpus and testing corpus. An example is as follows:

Testing corpus:[np-0-2 货币学派/n 及其/cC [np-0-1 政策/n 主张/n]]

Training corpus:[dj-1 [np-2 J·M·凯恩斯/nP 的/uJDE [np-1 政策/n 主张/n]] [vp-1 [pp-1 被/p [np-1 各/rB [np-1 主要/b [np-1 西方/nS 国家/n]]]] 采纳/v]]

“[np 政策/n 主张/n]” exists in both the above two sentences, but the core component contains two words in testing corpus, in contrast, the core component contains one word in training corpus with appearing five times in total. The core component likes this example will all be corrected. If such inconsistency can be avoided, the overall performance will be improved.

5 Experimental results and analysis

The training corpus and testing corpus of this paper are all provided by the CIPS-ParsEval-

2009 task 5. The experimental results of parsing tree are as follows:

Method	LP	LR	F1
CCRFs model based on MNP pre-processing	85.94%	85.99%	85.97%
CCRFs model based on MNP pre-processing + Local Optimization Method	86.04%	86.12%	86.08%
CCRFs model	84.47%	83.94%	84.2%
CCRFs model+Local Optimization Method	84.87%	84.57%	84.72%

Table 5. Experimental results of parsing

From table 5 we can see the F1-score of CCRF’s model based on MNP pre-processing method has reached 86.08%, higher 1.36% than not using this method. This indicates that the parsing method based on MNP pre-processing effectively reduces the parsing complexity and improves the performance. The two methods with local optimization increased the F-score by 0.11% and 0.52%, which indicates that local search methods are helpful for parsing.

In the case of MNP recognition accuracy is 100%, the results of MNP structure analysis and sentences frame analysis are shown in the following table:

Method	N1	N2	N3	P	R	F
MNP parsing based on CCRFs	31012	30999	26907	86.8%	86.76%	86.78%
Frame parsing based on CCRFs	31317	31355	28967	92.38%	92.5%	92.44%

Table 6. Parsing results of MNP and frame

	Average Length	len<5	5≤len<10	len≥10
MNP	5.33	5577	3097	1055
Sentence frame	5.16	4293	3174	774

Table 7. MNP & Sentence frame length distribution in testing corpus

We made experiments in the case of MNPs are all correctly identified and the F-score reached 89.4314%. Because the accuracy of MNP identification can not reach 100%, it seems that can be inferred that the F-score of statistical parsing based on MNP pre-processing can not be more than 89.4314%. But we can see from table 6 that the F-score of MNP internal structure analysis has reached 86.78%, only 0.7% higher than the sentence, but the F-score of sentence frame structure analysis has reached 92.44%. From table 7 we also can see that the complexity of MNP is roughly equal to the sentence frame, but the F-score of structural analysis has a large difference. This is because the ambiguity made by syntactic information described by the sentence as a unit is less than that by phrase as a unit. Therefore, the analysis of MNP structure can not be separated from the original sentence. But in this paper MNP structure analysis models are all trained by the MNP corpus without using the information of the original sentence. This shows that the F-score of MNP internal structure analysis has large potential to be improved and also shows that the results of statistical parsing based on MNP pre-processing are not entirely dependent on the effect of MNP recognition.

6 Conclusion

The paper presents a statistical parsing method of adding MNP pre-processing to statistical model. The paper only discusses the approach of MNP pre-processing added to CRF statistical model without performing detailed experiments for other statistical models. Our assumption is that adding MNP pre-processing to statistical model can effectively simplify the analysis of complex sentences and improve the parsing result comparing with simply using statistical models. Current experimental results indicate MNP pre-processing can really simplify the complex sentences, but the performance of MNP recognition and structural analysis needs to be improved, which will be our next work.

References

- Abney, S. 1991. *Parsing by chunks, Principle-Based Parsing*. Kluwer Academic Publishers.
- Black E., Jelinek F., Lafferty J., Magerman D. M., Mercer R. and Roukos S. 1992. Towards history-based grammars: Using richer models for probabilistic parsing. In *Proceedings of DARPA Speech and Natural Language Workshop*. Morgan Kaufmann.
- Brants T. 1999. Cascaded markov models. In *Proceedings of EACL'99*. Bergen, Norway.
- Cui Dai, Qiaoli Zhou, Dongfeng Cai, and Jie Yang. 2008. *Journal of Chinese Information Processing*, Vol 122, No. 6, pp. 110-115.
- Ejerhed, E. and Church, K. W. 1983. Finite state parsing. In *Proceedings of the Seventh Scandinavian Conference of Linguistics*. University of Helsinki, Finland.
- Erik Tjong Kim Sang. 2001. Transforming a chunker to a parser. In *Proceedings of Computational Linguistics in the Netherlands*. Rodopi, pp.177-188.
- John Lafferty, Andrew McCallum, Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th ICML*, pp. 282-289.
- Magerman, D.M. 1995. Statistical decision-tree models for parsing. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL'95)*. Cambridge, MA, USA.
- M. Collins. 1996. A Statistical Dependency Parser of Chinese Under Small Training Data. In *Proceedings of the 34th Annual Meeting of the ACL*. PP:184-191.
- M. Skounakis, M. Craven & S. Ray. 2003. Hierarchical Hidden Markov Models for Information Extraction. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, Acapulco, Mexico: Morgan Kaufmann, pp:427-433.
- Qiang Zhou, Maosong Sun & Changning Huang. 2000. Automatic Identification of Chinese Maximal Noun Phrase. *Journal of Software*, Vol 11, No. 2, pp. 193-201.
- Ratnaparkhi, A. 1998. Maximum Entropy Models for Natural Language Ambiguity Resolution, *PhD thesis Computer and Information Science*. University of Pennsylvania.
- Taku Kudo, Yuji Matsumoto. 2002. Japanese Dependency Analysis using Cascaded Chunking. In *Proceedings of CoNLL-2002*, pp:63-69.
- Yoshimasa Tsuruoka, Jun'ichi Tsujii. 2005. Chunk Parsing Revisited. In *Proceedings of the 9th international workshop on Parsing Technologies*. Vancouver, Canada. (IWPT 2005).