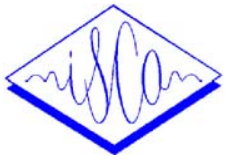
ISCA Archive
http://www.isca-speech.org/archive



INTERSPEECH 2004 - ICSLP
8th International Conference on Spoken Language Processing
ICC Jeju, Jeju Island, Korea
October 4-8, 2004

Partially Lexicalized Parsing Model Utilizing Rich Features

So-Young Park, Yong-Jae Kwak*, Joon-Ho Lim*, Hae-Chang Rim*, Soo-Hong Kim***

* Dept. of Computer Science Engineering, Korea University
{ssoya,yjkwak,jhlim,rim}@nlp.korea.ac.kr
** Dept. of Computer Software Engineering, Sangmyung University
soohkim@smu.ac.kr

10.21437/Interspeech.2004-244

Abstract

In this paper, we propose a partially lexicalized parsing model utilizing rich features to improve the parsing ability and reduce the parsing cost. In order to disambiguate parse trees effectively, it employs several useful features such as a syntactic label feature, a content feature, a functional feature, and a size feature. Besides, it is partially lexicalized so as to reduce the parsing cost closely connected with lexical information. Moreover, it is designed to be suitable for representing word order variation and constituent ellipsis in Korean sentences. Experimental results show that the proposed parsing model using more features performs better although it less depends on lexical information.

1. Introduction

Natural language parsing is regarded as a task of finding the appropriate parse tree for a given sentence. PCFG, a probabilistic approach, selects the best parse tree with the highest probability after estimating the probability of generating each parse tree according to CFG rules. However, PCFG is too simple to select the appropriate parse tree because it depends on the only CFG rules.

Let's look at one interesting example shown in Fig.1. In this example, a pos tagged noun phrase "*Pissa-n Seri-ui Os-dul*"(*"the expensive clothes of Seri"* in English) includes content morphemes such as a adjective (*adj*) "*Pissa*," a proper noun (*nnp*) "*Seri*," and a common noun (*noun*) "*Os*," and functional morphemes such as an adjective final ending (*afe*) "*-n*", an adjective case postposition (*acp*) "*-ui*", and a plural suffix (*suf*) "*-dul*". PCFG cannot discriminate between two parse trees of Fig.1 since the probability of generating the upper parse tree is equal to the probability of generating the lower parse tree. In this example, the rules (i) $NP \rightarrow$ [illegible] and (ii) $NP \rightarrow$ [illegible], and the rules (iii) $NP \rightarrow$ [illegible] and (iv) $NP \rightarrow$ [illegible] are equally treated.

In order to improve the syntactic disambiguation ability, most recent parsing models have been lexicalized [2, 4, 6, 10, 9] so that they can discriminate between (i) $SNP/$ [illegible] and (ii) $SNP/$ [illegible]. In addition, some of them also use

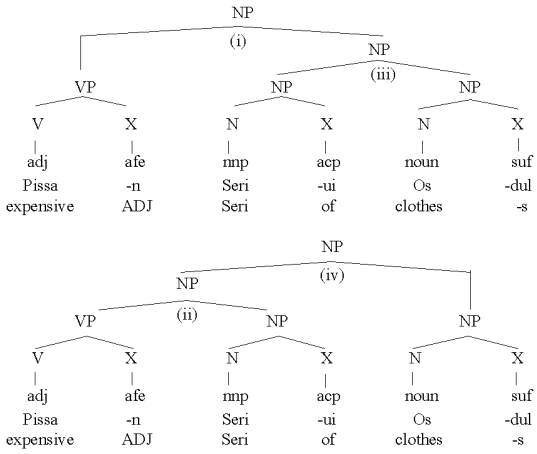


Figure 1: Two Parse Trees Generated by a CFG

the inside contexts [8, 9], the outside contexts [10] or the history [1, 2, 7]. However, the previous models cannot identify the structural difference between (iii) $SNP/$ [illegible] and (iv) $SNP/$ [illegible].

In this paper, we propose a method of utilizing rich features to distinguish between the rule (iii) and the rule (iv) as well as between the rule (i) and the rule (ii), and reducing the parsing cost related to lexical information. The rest of this paper is organized as follows: Section 2 explains the proposed partially lexicalized parsing model utilizing rich features, and Section 3 shows the experimental results. Finally, Section 4 concludes this paper.

2. Partially Lexicalized Parsing Model Utilizing Rich Features

Given a pos tagged sentence $w_1/p_1\ w_2/p_2\ \ldots\ w_k/p_k$, the best parse tree is selected based on the probability of generating a parse tree, which is calculated by multiplying the probabilities of all rules in the parse tree as follows:

$$T_{best}(w_1/p_1\ w_2/p_2\ \ldots\ w_k/p_k) = \underset{T}{argmax}\, P(T)$$

$$= \underset{T}{argmax} \prod_{i=1}^{k} P(n^{P_i} \rightarrow w_i/p_i) \prod_{i=k+1}^{j} P(n^{P_i} \rightarrow n^{L_i} n^{R_i})$$

where k is the number of unary rules, j is the number of all rules in the parse tree, n^{P_i} is the parent nonterminal of the i-th rule, n^{L_i} is its left child nonterminal, n^{R_i} is its right child nonterminal, w_i is the i-th word in a given sentence, and p_i is its pos tag.

As represented in the equation, the proposed parsing model restricts grammar rules to binary-oriented form so that they are classified into unary branching rules and binary branching rules. For the purpose of reducing grammar size without losing any grammar coverage, the n-ary branching rules are replaced by the combination of the binary branching rules. Moreover, this restriction is suitable for denoting Korean syntactic properties such as variable word order and constituent ellipsis.

In order to utilize enough information for effective syntactic disambiguation, the proposed parsing model represents a nonterminal $[l, c, f, s]$ as a syntactic label feature, a content feature, a functional feature, and a size feature. It can be rewritten as $[l, {}_wc/_pc, {}_wf/_pf, {}_as/_bs]$ as illustrated in Fig.2. A syntactic label feature indicates a syntactic label such as NP or VP. A content feature is used for describing the content of a nonterminal where ${}_wc/_pc$ indicates the content head word and its pos tag. A functional feature represents the function of the nonterminal where ${}_wf/_pf$ indicates the functional head word and its pos tag. A size feature describes how complex a nonterminal is where ${}_as/_bs$ indicates the number of eojeols (Korean spacing unit) covered by a nonterminal, and its size tag such as S (*small*) satisfying $1 \leq {}_as \leq 3$, M (*medium*) fulfilling $4 \leq {}_as \leq 6$, or L (*large*).

Particularly, a content feature ${}_wc/_pc$ and a functional feature ${}_wf/_pf$ are partially lexicalized according to word frequency so as to alleviate sparse data and reducing the parsing cost related to enormous lexicalized rules. Since the low-frequent words cannot be guaranteed to estimate the proper probabilities, the content word feature ${}_wc$ and the functional word feature ${}_wf$ take either a word itself representing high-frequent words, or a null tag "*" replacing low-frequent words according to the assumption that the low-frequent words belonging to the identical pos have the same syntactic properties.

For example, Fig.2 shows the other binary branching parse trees for the noun phrase "*Pissa-n Seri-ui Os-dul*" based on the assumption that a proper noun "*Seri*" and a plural suffix "*-dul*" are replaced by a null tag "*" in the content word feature ${}_wc$ or the functional word feature ${}_wf$. A nonterminal $[NP, Os/noun, */suf, 1/S]$ describes a word "*Os-dul*". Its syntactic label feature indicates NP like Fig.1, and its content feature "*Os/noun*" indicates the content morpheme "*Os*". Its functional feature describes that a null tag "*" is substituted for the final functional morpheme "*-dul*". And then, its size feature indicates a single eojeol. The proposed parsing model can discriminate between (i) $P($ $[NP, Os/noun, */suf, 3/S] \rightarrow [VP, Piss/adj, -n/afe, 1/S]$ $[NP, Os/noun, */suf, 2/S]$ $)$ and (ii) $P($ $[NP, */nnp, -ui/acp, 2/S] \rightarrow [VP, Piss/adj, -n/afe, 1/S]$ $[NP, */nnp, -ui/acp, 1/S]$ $)$. In addition, (iii) $P($ $[NP, Os/noun, */suf, 2/S] \rightarrow [NP, */nnp, -ui/acp, 1/S]$ $[NP, Os/noun, */suf, 1/S]$ $)$ is different from (iv) $P($ $[NP, Os/noun, */suf, 3/S] \rightarrow [NP, */nnp, -ui/acp, 2/S]$ $[NP, Os/noun, */suf, 1/S]$ $)$.

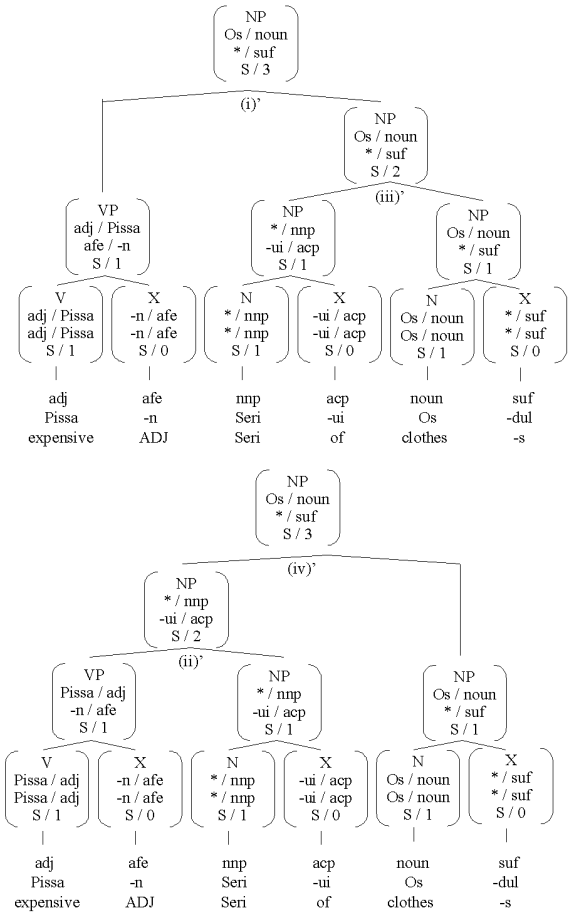


Figure 2: Two Parse Trees Generated by FKG

As compared with a syntactic label NP, the proposed nonterminal such as $[NP, Os/noun, */suf, 1/S]$ or $[NP, Os/noun, */suf, 2/S]$ can provide more information for effective syntactic disambiguation. Nevertheless, every parse tree including these features such as Fig.2 can be generated from a CFG-style parse tree such as Fig.1 without any resource; since all features can be extracted from the CFG-style parse tree with the pos tagged words.

3. Experimentation

To evaluate parsing performance according to feature combination and the amount of lexicalization, we assign some selected feature to a nonterminal as shown in each line of Fig.3 while we allow a content feature ${}_wc$ and a functional feature ${}_wf$ to take either a null tag "*" or a

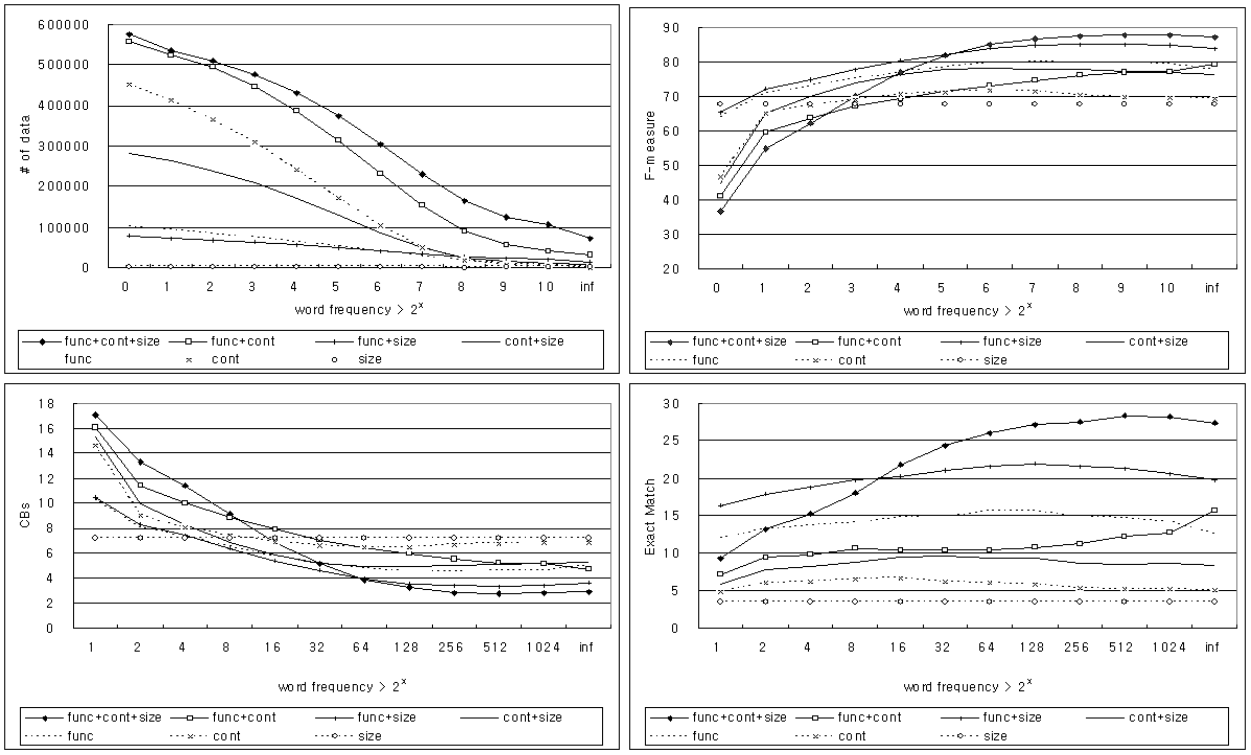


Figure 3: Parsing Performance according to Feature Combination and the Amount of Lexicalization

head word satisfying $word\ frequency \geq 2^x$ as shown in X-axis of Fig.3. A treebank[1] are divided into 90% for the training set and 10% for the test set. And then, we measure data size, F-measure, cross brackets, and exact matching of the model using the feature combination [5].

Fig.3 shows that a nonterminal including more features generally makes the proposed parsing model perform better. It is remarkable that a single functional feature is more useful than the combination of a content feature and a size feature. Besides, the combination of different features such as a functional feature and a size feature is preferred to the combination of similar features such as a functional feature and a content feature. The reason is that the effect of the functional feature may overlap the effect of the content feature because the former represented by a word and its pos is the same as the latter. On the other hand, we can find that the parsing model using more features can less depend on lexical information. Also, the less lexicalized parsing model utilizes the less number of data while the more lexicalized parsing model is more influenced by sparse data.

4. Conclusion

In this paper, we propose a partially lexicalized parsing model utilizing rich features such as a syntactic label feature, a content feature, a functional feature, and a size feature. Besides, it allows the content feature and the functional feature to replace a null tag "*" for a low-frequent word in order to reduce the expensive parsing cost related to lexical information. Experimental results show that the parsing model using more features performs better although it less depends on lexical information. Particularly, the combination of different features is preferred to the combination of similar features. Also, it is remarkable that the functional feature is more useful than the combination of the content feature and the size feature. For the future work, we will study on a new selective lexicalization method in order to improve parsing performance.

[1] KAIST treebank [3] of 31,080 Korean sentences is transformed into the binary branching style such as Fig.1.

5. Acknowledgements

This work was supported by Korea Research Foundation Grant (KRF-2002-042-D20485).

6. References

[1] Black, E., F. Jelinek, J. Lafferty, David M. Magerman, R. Mercer, and S. Roukos. 1993. Towards History-based Grammars: Using Richer Models for Probabilistic Parsing *In Proceedings of In Proceedings of ACL*, pp.31–37.

[2] Charniak, Eugene. 2001. Immediate-Head Parsing

for Language Models. *In Proceedings of ACL*, 116–123.

[3] Choi, Key-Sun. 2001. KAIST Language resources v.2001. *Result of Core Software Project from Ministry of Science and Technology, Korea*, (http://kibs.kaist.ac.kr)

[4] Collins, Michael. 1999. Head-Driven Statistical Models for Natural Language Parsing. *Ph.D. Thesis, University of Pennsylvania*.

[5] Goodman, Joshua. 1996. Parsing Algorithms and Metrics. *In Proceedings of In Proceedings of ACL 1996*, pp.177–183.

[6] Goodman, Joshua. 1997. Probabilistic Feature Grammars. *In Proceedings of the 5th International Workshop on Parsing Technologies*, pp.89–100.

[7] Klein, Dan, and Christopher D. Manning. 2003. Accurate Unlexicalized Parsing. *In Proceedings of ACL*, 423-430.

[8] Kwak, Yong-Jae, Young-Sook Hwang, Hoo-Jung Chung, So-Young Park, Sang-Zoo Lee, and Hae-Chang Rim. 2001. GLR Parser with Conditional Action Model(CAM). *In Proceedings of NLPRS*, pp.359–366.

[9] Magerman, David M. 1995. Statistical Decision-Tree Models for Parsing. *In Proceedings of ACL*, pp.276–283.

[10] Lee, Kong-Joo, Jae-Hoon Kim, and Kil-Chang Kim. 1997. An Efficient Parsing of Korean Sentences Using Restricted Phrase Structure Grammar. *Journal of Computer Processing of Oriental Languages*, 11(1):49–62.