Proceedings of the IASTED International Conference
ARTIFICIAL INTELLIGENCE AND SOFT COMPUTING
July 24-26, 2000, Banff, Canada

A SYNTACTIC ANALYSIS METHOD USING INDUCTIVE LEARNING

YOSHIYUKI MASATOMI, KENJI ARAKI and KOJI TOCHINAI
Graduate School of Engineering,
Hokkaido University
N13 W8, Kita-ku,
Sapporo, Japan, 060-8628
(tome, araki, tochinai)@media.eng.hokudai.ac.jp

ABSTRACT

In this paper, we describe a syntactic analysis method using inductive learning, as well as details concerning the experiment used in the performance evaluation for this method. In our proposed method, the parser acquires parsing rules using the examples of parsing results, and it parses Japanese sentences using the acquired parsing rules. We consider our proposed method can resolve problems associated with the Rule-based method, the Example-based method and statistical method. We performed the experiment using EDR Japanese Corpus. In this study, we evaluate the results and the ability of using our proposed method of inductive learning.

Keywords: Syntactic analysis, Inductive learning, Parser, Dependency

1 INTRODUCTION

In natural language processing, parsing is very important in analyzing sentences. Therefore, a great deal of research concerning the syntactic analysis method has been focused on. The main method of parsing is the Rule-based method [1][2]. However, the Rule-based method cannot deal adequately with various linguistic phenomena due to its use of limited rules. Therefore, its parsing ability is low.

To resolve this problem, Example-based approach has become a common technique for natural language processing, especially in machine translations [3][4][5]. Example-based parser was recently proposed [6]. The correct parsing rate and the accuracy of Example-based parser is high because this method adapts to data automatically. However, this method requires many examples of parsed sentences in order to parse sentences correctly.

Moreover, the syntactic analysis methods which utilize tree structure were proposed [7][8][9]. These methods parse sentences stochastically by using a tree structure. In addition to the methods mentioned above, a method to extract grammar from a corpus automatically has been proposed [10]. However, the ability of the parser based on statistical method depends on sample sentences.

In order to resolve the problems associated with these syntactic analysis methods, we proposed a syntactic analysis method using inductive learning. In this method, parsing rules are inductively acquired and parsing results are produced through the use of these rules. This method does not decide the solution stochastically. We consider it is possible to acquire parsing rules automatically, and extract parsing rules of various abstraction degrees from the acquired rules recursively. In our proposed method, parsing rules are not grammatical. These rules are extracted from word strings by using inductive learning. Moreover, we consider that the parser based on our proposed method can improve accuracy by applying parsing rules of the most suitable abstraction degree to the input word string. A parser based on this method will continuously evolve to higher level of quality, and will actively adapt to the input data. Because the learning algorithm does not depend on the language in this method, we believe it can be applied to various languages.

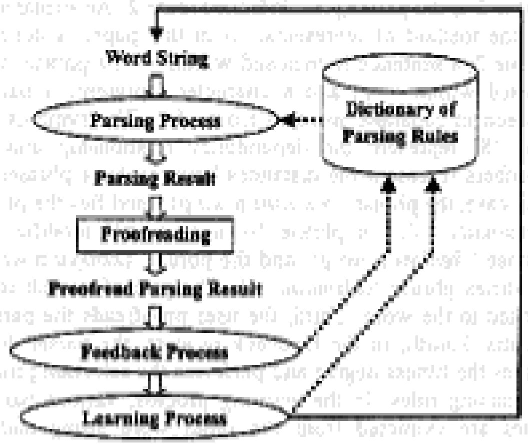


Figure 1: Process

2 OUTLINE

Figure 1 shows the outline of this method. The experimental system based on this method is a Japanese parser.

First, the user inputs a word string (in our case, a Japanese sentence). This word string is obtained by recognizing the word by using inductive learning [11]. Table 1 shows an example of a Japanese sentence and its word string. In this paper, Japanese are written in *Italics*. Second, in the parsing process, the parser produces parsing results using the parsing rules which has already ac-

Table 1: Example of a Japanese sentence

Japanese	*watashiwaokunokenkyuwookonatta.*						
English	I did many kinds of research.						
Word string	*watashi*	*wa*	*oku*	*no*	*kenkyu*	*wo*	*okonatta.*

Table 2: Example of representation of parsing result in this system

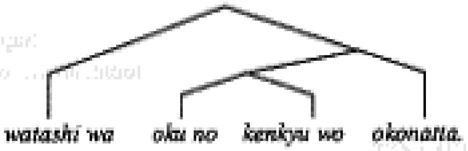
Parsing result	[(*watashi* n *wa* p)#3 (*oku* n *no* p)#1 $1(*kenkyu* n *wo* p)#1 $1$3(*okonatta* v .)]

[ ]: Sentence, ( ): Phrase, #,$: Dependency relationship, n: Noun, p: Particle, v: Verb

results using the parsing rules which has already acquired in the learning process. Figure 2 shows an example of a parse tree. Table 2 shows an example of the representation of the parsing result from this parser. The representation in Table 2 is the parsing result from Figure 2. An explanation of the method of representation in this paper is done. In Table 2, a sentence is enclosed within '[ ]', a phrase is enclosed within '( )', and a character represents a part of speech of Japanese word in front of it. The symbols: '#' and '$' represent the dependency relationship and the numbers represent the distances between their phrases. In this case, the phrase '(*watashi* n *wa* p)' modifies the phrase '(*okonatta* v)', the phrase '(*oku* n *no* p)' modifies the phrase '(*kenkyu* n *wo* p)' and the phrase '(*kenkyu* n *wo* p)' modifies phrase '(*okonatta* v)'. The part of speech is attached to the word. Third, the user proofreads the parsing results. Fourth, in the feedback process, the parser determines the fitness degree and performs the selection process of parsing rules. In the learning process, various parsing rules are extracted from the input word string and its proofreading parsing result by inductive learning. Table 3 shows examples of parsing rules. There are three kinds of parsing rules, 'sentence parsing rule', 'dependency parsing rule' and 'word parsing rule'. In this table, the words to the left of the colon ':' are word strings and, the words to the right of the colon ':' are representations of the parsing results in this parser. Moreover, new parsing rules are recursively extracted from the acquired parsing rules in this process. The dictionary of parsing rules becomes more defined or exact by repetition of the above-mentioned processes. Thus, the parser continuously improves its quality of parsing.

2.1 PARSING PROCESS

In this process, the parser parses the input word strings. The best match parsing rule is sequentially applied to the input word string and a parsing result is produced. Figure 3 shows an example of how the parser produces a parsing result. In this figure, (1)word string input. (2)The parser retrieves the best match parsing rules from the dictionary of parsing rules. (3)The parser applies the retrieved parsing rules to the input word string. In this case, the asterisk '*' is an unknown dependency relationship. This dependency relationship is analyzed by the dependency parsing rule (b). (4)The parser outputs the parsing result. When two or more parsing rules can be applied, the rule is applied in the following order.



(I did many kinds of research.)

Figure 2: Example of parse tree

- The match rule is the longest and maximum number of characters.
- The number of rules composed is minimum.
- The average of correct parsing rate of the parsing rules which compose is the highest one.
- The total of correct parsing degree of the parsing rules which compose is the highest one.
- The total of error-parsing degree of the parsing rules which compose is the lowest one.
- The dictionary procedure is first.

2.2 FEEDBACK PROCESS

The procedure for the feedback process is as follows.

- The parser decides the common parts and the different parts between the parsing result and the proofreading parsing result.
- The common parts are assumed correct.
- The different parts are assumed incorrect.
- The parser adds 1 to the correct parsing frequency of the correctly analyzed parsing rules.
- The parser adds 1 to the error-parsing frequency of the erroneously analyzed parsing rules.

2.3 LEARNING PROCESS

In the learning process, the parsing rules are acquired by using the word strings and the proofreading parsing result. The parsing rules are acquired from the common parts and the different parts. The parser recursively repeats this process until a new rule is not extracted from the acquired parsing rules [12]. Then, the dependency relationship is assumed that it is a common part only when two or more relations between all words are the same. The parsing rules concerning the dependency relationship are acquired from the different parts and, the parser retains the parsing rules of the dependency relationship. The parser acquires the parsing rules when the common part rate of the dependency relationship is 50% or more and the character corresponding rate is 50% or more.

Table 3: Examples of parsing rules

Sentence parsing rule	<watashi wa oku no kenkyu wo okonatta . : [(watashi n wa p)#3 (oku n no p)#1 $1(kenkyu n wo p)#1 $1$3(okonatta v .)]> <@0 wa oku no @1 wo @2 . : [(@0 wa p)#3 (oku n no p)#1 $1(@1 wo p)#1 $1$3(@2 .)]>
Dependency parsing rule	/(watashi wa) (okonatta .) : (watashi n wa p) (okonatta v .)/ /(oku no) (kenkyu wo) : (oku n no p) (kenkyu n wo p)/ /(kenkyu wo) (okonatta .) : (kenkyu n wo p) (okonatta v .)/ /(@0 wa) (okonatta .) : (@0 wa p) (okonatta v .)/ /(oku no) (@0 wo) : (oku n no p) (@0 wo p)/
Word parsing rule	<watashi : watashi n> <kenkyu : kenkyu n> <okonatta : okonatta v>

[ ]: Sentence, ( ): Phrase, #,$: Dependency relationship, n: Noun, p: Particle, v: Verb, @: Variable

```
(1)  The input word string:
        watashi wa oku no kenkyu wo okonatta .
      • •      (I did many kinds of research.)
(2)  The parsing rules:
      (a)   <@0 wa oku no @1 wo @2 . : [(@0 wa p)*0 (oku n no p)#1 $1(@1 wo p)#1 $1(@2 .)]>
      (b)   /(watashi wa) (okonatta .) : (watashi n wa p) (okonatta v .)/
      (c)   <watashi : watashi n>    <kenkyu :kenkyu n>    <okonatta : okonatta v>
(3)  The application of parsing rules:
   •      (a) and (c)
          <watashi wa oku no kenkyu wo okonatta . :
                      [(watashi n wa p)*0 (oku n no p)#1 $1(kenkyu n wo p)#1 $1(okonatta v .)]>
   •      (b)
          <watashi wa oku no kenkyu wo okonatta . :
                   [(watashi n wa p)#3 (oku n no p)#1 $1(kenkyu n wo p)#1 $1$3(okonatta v .)]>
(4)  The parsing result:
                   [(watashi n wa p)#3 (oku n no p)#1 $1(kenkyu n wo p)#1 $1$3(okonatta v .)]
```

Figure 3: Example of parsing process

Figure 4 shows an example of the extracted parsing rules from the acquired parsing rules using inductive learning. In this figure, there are the acquired parsing rules (a)~(d) in the dictionary of parsing rules. The rules (a) and (b) are sentence parsing rules. The rules (c) and (d) are dependency parsing rules. In (a) and (b), the underlined parts are the different parts. The common parts do not require changing, and the parser change the different parts into variables. Thus, a new parsing rule (e) is extracted. Thereby, new word parsing rules (f) are extracted. In the same way, a new dependency parsing rule (g) is extracted from (c) and (d). Thereby, new word parsing rules are extracted from the different parts.

3 EXPERIMENT

We performed the experiment using EDR Japanese Corpus [13]. In this experiment, we used 5,000 sentences in the corpus. A single sentence is input to the parser. That is, input sentences are unknown. Therefore, the parser can dynamically adapt to the input sentences.

In this experiment, we concentrated only on the evaluation of the dependency relationship. Because, the number of sentences used in this experiment was not enough to fully demonstrate the ability of the parser substantially. We consider the ability of the parser can be demonstrated only if the number of sentences is greatly increased.

In this experiment, a correct parsing result defined the same type of dependency relationship as the EDR Japanese Corpus.

3.1 EXPERIMENT PROCEDURE

The dictionary of parsing rule is empty in the initial state. We experimented using the following procedures.

I. A Japanese sentence is changed into a word string.
II. The word string is input.
III. The parser parsed the input word string with the dictionary of parsing rules and a parsing result is obtained. (In this experiment, the result is the dependency relationship between phrases.)
IV. The feedback processing is done from the parsing result and EDR Japanese corpus.
V. The parsing rules are acquired by using inductive learning from pairs in the word string and the parsing result that has been proofread.
VI. New parsing rules are extracted from the parsing rules that have already been acquired using inductive learning in step V.
VII. Step VI is recursively repeated until no new rules are extracted.

(1) The acquired parsing rules

(a) <@0 *wa kenkyu wo okonatta* . : [(@0 *wa* p)#2 (*kenkyu* n *wo* p)#1 $1$2(*okonatta* v .)]>

(b) <@0 *wa katsudo wo okonatta* . : [(@0 *wa* p)#2 (*katsudo* n *wo* p)#1 $1$2(*okonatta* v .)]>

(c) /(*watashi wa*) (*okonatta* .) : (*watashi* n *wa* p) (*okonatta* v .)/

(d) /(*kare wa*) (*okonatta* .) : (*kare* n *wa* p) (*okonatta* v .)/

(2) The extracted parsing rules

From (a) and (b)

(e) <@0 *wa* @1 *wo okonatta* . : [(@0 *wa* p)#2 (@1 *wo* p)#1 $1$2(*okonatta* v .)]>

(f) <*kenkyu* : *kenkyu* n> <*katsudo* : *katsudo* n>

From (c) and (d)

(g) /(@0 *wa*) (*okonatta* .) : (@0 *wa* p) (*okonatta* v .)/

(h) <*watashi* : *watashi* n> <*kare* : *kare* n>

Figure 4: Example of the extracted parsing rules from the acquired parsing rules

VIII. The operation above is repeated using about 5,000 Japanese sentences.

In step V, the number of sentences that the parser learns is 100 sentences before the input sentence. Because, we consider that it is unnecessary to learn all sentences from the standpoint of human learning research.

3.2 EXPERIMENT RESULTS

Figure 5 shows the change of the correct parsing rate for this experiment. The total correct parsing rate was 22.7%. The correct parsing rate between 2,501 and 3,000 sentences was 28.0%. Moreover, the correct parsing rate between 4,501 and 5,000 sentences was 33.9%. The correct parsing rate of the parser based on our proposed method is slightly higher. Therefore, we confirm that the functionality of this method is high and effective.

4 DISCUSSION

In Figure 5, the accuracy of the parsing results based on this method improves gradually. In the initial state, the dictionary of parsing rules is empty. Therefore, when the number of input sentences is low, the number of parsing rules is also low. Then, the parsing rules that can be applied to the input word string are not usually in the dictionary of parsing rules. Therefore, the dependency relationship cannot be determined during the parsing process, and the correct parsing rate is low when the number of input sentences is low. However, the number of parsing rules increases when the number of input sentences increases, and the number of parsing rules which can be applied to the input word string increases. As a result, the dependency relationship can be determined during the parsing process and the correct parsing rate rises, as well. Therefore, our proposed method has the ability to improve accuracy of the dependency relationship gradually and increased effectiveness can be realized.

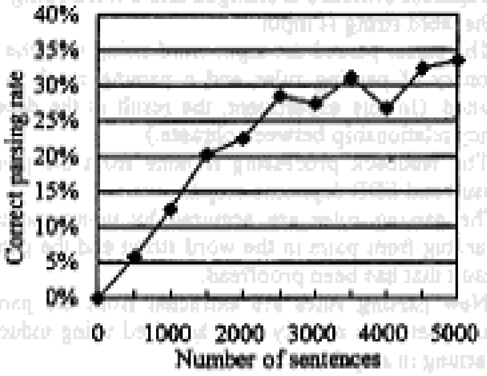


Figure 5: Change of parsing rule

Next, we describe an example of the error parsing in the experiment. In the case that the parser applied an acquired dependency parsing rule to an input word string, the parser also applied an error-parsing rule to unrelated phrases. An error occurred often when the variable was included in the parsing rule. We consider the error occurred because the parser applied the parsing rules as often as possible. However, we also consider the number of errors will gradually decrease as the number of parsing rules increases.

In the case in which the number of parsing rules increases, a new problem occurs, increase of processing time. In the parsing process and learning process, the processing time increases. To decrease time in the parsing process, we will examine a classification of the dictionary of parsing rules. The time to find the applied parsing rule in the dictionary can be decreased during the parsing process. Moreover, to decrease time during the learning process, the parser should extract the parsing rules efficiently. Therefore, we examine the addition of heuristics.

5 COMPARISON WITH OTHER METHODS

In this chapter, we describe the comparison between our proposed method and the other methods.

As described in Chapter 1, the main method of a parser is the Rule-based method [1][2]. In this method, the grammatical rules must be manually prepared. Therefore, this method cannot deal adequately with various linguistic phenomena due to its use of limited rules. Thus, its correct parsing rate is low and the quality is poor. In our proposed method, the parsing rules can be automatically acquired from the examples of parsing by using inductive learning. As a result, a parser based on our proposed method can dynamically adapt to the input sentences and deal adequately with various linguistic phenomena.

Next, we attempt to compare our proposed method with the Example-based method [6]. In the Example-based method, the input sentence is compared to examples, the examples which can be applied are selected and the parsing results are output. That is, the parsing results depend on the selected examples of parsing. Therefore, a huge amount of examples is required in order to raise the accuracy of parsing. In our proposed method, the parser automatically makes the abstracted parsing rules. Its rules are extracted from the relationships between examples and have the best match abstraction degree. That is, the parsing rules are recursively extracted from the examples of parsing which has already been acquired. At this time, inductive learning is applied. As a result, we consider that the parsing rules can be effectively extracted from a smaller number of examples of parsing than the amount required for the Example-based method.

In addition, we attempt to compare our proposed method with the statistical method [7][8][9]. In the statistical method, the sample examples of parsing are necessary in order to obtain a likely solution. In a case in which the input sentence is a similar type as the sample examples of parsing, it is a very effective method. However, there is the problem in which the ability of the parser when based on statistical method depends on sample sentences. Moreover, when the parser is based on statistical method, it cannot deal with an unknown input sentence in the sample sentences or a change in topics. In our proposed method, the parser acquires new rules to unknown input sentences and it is possible to dynamically adapt to a change of topics. This is because, the sentences are input individually, and inductive learning is applied to each sentence in our proposed method.

Therefore, we believe that our proposed method has a high parsing ability compared with the other methods.

6 CONCLUSIONS

To resolve the problems which the Rule-based method, Example-based method and statistical method have, we proposed a syntactic analysis method which utilizes inductive learning. In addition, we designed a parser based on our proposed method. In the parser, parsing rules were acquired by using inductive learning, and recursively extracted from the parsing rules that had been acquired. As a result, the parsing rules could be acquired automatically, and the parser acquired the parsing rules of a variety of abstractions.

Moreover, we evaluated the performance of this parser using EDR Japanese Corpus. In this experiment, the parser applied the parsing rules of the most suitable abstraction degree to the input word strings. We confirmed that the correct parsing rate increased gradually. In addition, we attempted to compare our proposed method with the other methods in Chapter 5. Therefore, we concluded that the ability of this method is high and an effective method in parsing.

To improve accuracy, we will examine the addition of heuristics and a classification of the dictionary of parsing rules, etc. We continue this study using a large variety of Japanese sentences. We also would like to apply this method to other languages as well.

REFERENCES

[1] E.Brill and P.Resnik, A rule-based approach to prepositional phrase attachment disambiguation, *Proceedings of the 15th COLING*, 1994, 1198-1204.

[2] U.Germann, A deterministic dependency parser for Japanese, *Asia-Pacific Association for Machine Translation*, 1999, 547-555.

[3] V.Sadler and Vendelmans, Pilot implementation of a bilingual knowledge bank, *Proceedings of the 13th COLING*, 1990, 449-451.

[4] C.Stanfill and D.Waltz, Toward memory-based reasoning, *Communications of the ACM*, 29(12), 1986, 1213-1228.

[5] S.Sato and M.Nagao, Toward memory-based translation, *Proceedings of the 13th COLING*, 1990, 247-252.

[6] M.Al-Adhaileh, and T.E.Kong, A flexible example-based parser based on the SSTC, *Proceeding of the ACL*, 1998, 687-693.

[7] M.Haruno, S.Shirai, and Y.Ooyama, A Japanese dependency parser based on a decision tree, *Transactions of Information Processing Society of Japan*, 39(12), 1998, 3177-3186.

[8] R.Bod, A computational model of language performance data oriented parsing, *Proceedings of the 14th COLING*, 1992, 855-859.

[9] S.Mori and M.Nagao, A stochastic language model using dependency, *IEICE Technical Report*, NL122-6, 1997.

[10] K.Shirai, T.Tokunaga, and H.Tanaka, Automatic extraction of Japanese probabilistic context free grammar from a bracketed corpus, *Transactions of Natural Language Processing Society of Japan*, 4(1), 1997, 125-146.

[11] K.Araki, and K.Tochinai, Acquisition words by inductive learning and recognition words using certainty, *Transactions of Institute of Electronics, Information and Communication Engineers*, J75-D-II(7), 1992, 1213-1221.

[12] K.Araki, Y.Takahashi, Y.Momouchi, and K.Tochinai, Non-segmented kana-kanji translation using inductive learning, *Transactions of Institute of Electronics, Information and Communication Engineers*, J79-D-II(3), 1996, 391-402.

[13] The EDR Corpus (Japan Electronic Dictionary Research Institute).