# Japanese Morphological Analyzer using Word Co-occurrence
# — JTAG —

Takeshi FUCHI
NTT Information and Communication Systems Laboratories
Hikari-no-oka 1-1
Yokosuka 239-0847, Japan,
fuchi@isl.ntt.co.jp

Shinichiro TAKAGI
NTT Information and Communication Systems Laboratories
Hikari-no-oka 1-1
Yokosuka 239-0847, Japan,
takagi@nttnly.isl.ntt.co.jp

## Abstract

We developed a Japanese morphological analyzer that uses the co-occurrence of words to select the correct sequence of words in an unsegmented Japanese sentence. The co-occurrence information can be obtained from cases where the system incorrectly analyzes sentences. As the amount of information increases, the accuracy of the system increases with a small risk of degradation. Experimental results show that the proposed system assigns the correct phonological representations to unsegmented Japanese sentences more precisely than do other popular systems.

## Introduction

In natural language processing for Japanese text, morphological analysis is very important. Currently, there are two main methods for automatic part-of-speech tagging, namely, corpus-based and rule-based methods. The corpus-based method is popular for European languages. Samuelsson and Voutilainen (1997), however, show significantly higher achievement of a rule-based tagger than that of statistical taggers for English text. On the other hand, most Japanese taggers[1] are rule-based. In previous Japanese taggers, it was difficult to increase the accuracy of the analysis. Takeuchi and Matsumoto (1995) combined a rule-based and a corpus-based method, resulting in a marginal increase in the accuracy of their taggers. However, this increase is still insufficient. The source of the trouble is the difficulty in adjusting the grammar and parameters. Our tagger is also rule-based. By using the co-occurrence of words, it reduces the difficulty and generates a continuous increase in its accuracy.

The proposed system analyzes unsegmented Japanese sentences and segments them into words. Each word has a part-of-speech and phonological representation. Our tagger has the co-occurrence information of words in its dictionary. The information can be adjusted concretely by hand in each case of incorrect analysis. Concrete adjustment is different from detailed adjustment. It must be easy to understand for people who make adjustments to the system. The effect of one adjustment is concrete but small. Therefore, much manual work is needed. However, the work is so simple and easy.

Section 1 shows the drawbacks to previous systems. Section 2 describes the outline of the proposed system. In Section 3, the accuracy of the system is compared with that of others. In addition, we show the change in the accuracy while the system is being adjusted.

## 1 Previous Japanese Morphological Analyzers

Most Japanese morphological analyzers use linguistic grammar, generate possible sequences of words from an input string, and select a sequence. The following are methods for selecting the sequence:

- Choose the sequence that has a longer word on the right-hand side. (right longest match principle)

[1] In this paper, a tagger is identical to a morphological analyzer.

- Choose the sequence that has a longer word on the left-hand side. (left longest match principle)
- Choose the sequence that has the least number of phrases. (least number of phrases principle)
- Choose the sequence that has the least connective-cost of words. (least connective-cost principle)
- Use pattern matching of words and/or parts-of-speech to specify the priority of sequences.
- Choose the sequence that contains modifiers and modifiees.
- Choose the sequence that contains words used frequently.

In practice, combinations of the above methods are used.

Using these methods, many Japanese morphological analyzers have been created. However, the accuracy cannot increase continuously in spite of careful manual adjustments and statistical adjustments. The cause of incorrect analyses is not only unregistered words, in fact, many sentences are analyzed incorrectly even though there is a sufficient vocabulary for the sentences in their dictionaries. In this case, the system generates a correct sequence but does not select it. Parameters such as the priorities of words and connective-costs between parts-of-speech, can be adjusted so that the correct sequence is selected. However, this adjustment often causes incorrect side effects and the system analyzes other sentences incorrectly that have already been analyzed correctly. This phenomenon is called 'degrading'.

In addition to parameter adjustment, parts-of-speech may need to be expanded. Both operations are almost impossible to complete by people who are not very familiar with the system. If the system uses a complex algorithm to select a sequence of words, even the system developer can hardly grasp the behaviour of the system.

These operations begin to become more than what a few experts can handle because vocabularies in the systems are big. Even to add an unregistered word to a dictionary, operators must have good knowledge of parts-of-speech, the priorities of words, and word classification for modifiers and modifiees. In this situation, it is difficult to increase the number of operators. This is situation with previous analyzers.

Unfortunately, current statistical taggers cannot avoid this situation. The tuning of the systems is very subtle. It is hard to predict the effect of parameter tuning of the systems. To avoid this situation, our tagger uses the co-occurrence of words whose effect is easy to understand.

## 2 Overview of our system

We developed the Japanese morphological analyzer, JTAG, paying attention to simple algorithm, straightforward adjustment, and flexible grammar.

The features of JTAG are the followings.

- An attribute value is an atom.

In our system, each word has several attribute values. An attribute value is limited so as not to have structure. Giving an attribute value to words is equivalent to naming the words as a group.

- New attribute values can be introduced easily.

An attribute value is a simple character string. When a new attribute value is required, the user writes a new string in the attribute field of a record in a dictionary.

- The number of attribute values is unlimited.
- A part-of-speech is a kind of attribute value.
- Grammar is a set of connection rules.

Grammar is implemented with connection rules between attribute values. List 1 is an example[2].

One connection rule is written in one line. The fields are separated by commas. Attribute values of a word on the left are written in the first field. Attribute values of a word on the right are written in the second field. In the last field, the cost[3] of the rule is written. Attribute values are separated by colons. A minus sign '-' means negation.

For example, the first rule shows that a word with 'Noun' can be followed by a word with

| | | |
|---|---|---|
| Noun, | Case:ConVerb, | 50 |
| Noun:Name, | Postfix:Noun, | 100 |
| Noun:-Name, | Postfix:Noun, | 90 |
| Copula:de, | VerbStem:Lde, | 50 |

**List 1: Connection rules.**

[2] Actual rules use Japanese characters.

[3] The cost figures were intuitively determined. The grammar is used mainly to generate possible sequences of words, so the determination of the cost figures was not very subtle. The precise selection of the correct

| | JTAG | JUMAN | CHASEN |
|---|---|---|---|
| Vocabulary | 350K | 710K | 115K |
| Standard Words | 11809 | 9830 | 9901 |
| Output Words | 11855 | 9864 | 9948 |
| Segmentation | 98.9% \| 99.3% | 98.9% \| 99.3% | 98.5% \| 98.9% |
| Segmentation & Part-of-Speech | 98.8% \| 99.2% | 98.3% \| 98.7% | 97.6% \| 98.1% |
| Segmentation & Phoneme | 98.8% \| 99.2% | 98.2% \| 98.6% | 97.5% \| 97.9% |
| **Segmentation & Phoneme & Part-of-Speech** | **98.7% \| 99.1%** | **98.0% \| 98.3%** | **97.1% \| 97.6%** |

**Table II: Accuracy per word (precision | recall)**

'Case' and 'ConVerb'. The cost of the rule is 50.

The second rule shows that a word with 'Noun' and 'Name' can be followed by a word with 'Postfix' and 'Noun'. The cost is 100. The third rule shows that a word that has 'Noun' and does not have 'Name' can be followed by a word with 'Postfix' and 'Noun'. The cost is 90.

Only the word 'で' has the combination of 'Copula' and 'de', so the fourth rule is specific to 'で'.

- The co-occurrence of words.

In our system, the sequence of words that includes the maximum number of co-occurrence of words is selected. Table I shows examples of records in a dictionary.

'額' means 'amount', 'frame', 'forehead' or a human name 'Gaku'. In the co-occurrence field, words are presented directly. If there are no co-occurrence words in a sentence that includes '額', 'amount' is selected because its cost is the smallest. If '絵'(picture) is in the sentence, 'frame' is selected.

- Selection Algorithm

JTAG selects the correct sequence of words using connective-cost, the number of co-occurrences, the priority of words, and the length of words. The precise description of the algolithm is shown in the Appendix.

This algolithm is too simple to analyze Japanese sentences perfectly. However, it is sufficient in practice.

sequence is done by the co-occurrence of words.

# 3 Evaluation

In this section, Japanese morphological anayzers are evaluated using the following :

- Segmentation
- Part-of-speech tagging
- Phonological representation

JTAG, is compared with JUMAN[4] and CHASEN[5]. A single "correct analysis" is meaningless because these taggers use different parts-of-speech, grammars, and segmentation policies. We checked the outputs of each and selected the incorrect analyses that the grammar maker of each system must not expect.

## 3.1 Comparison

To make the output of each system comparable, we reduce them to 21 parts-of-speech and 14 verb-inflection-types. In addition, we assume that the part-of-speech of unrecognized words is Noun.

The segmentation policies are not unified. Therefore, the number of words in sentences is different from each other.

Table II shows the system accuracy. We used 500 sentences[6] (19,519 characters) in the EDR[7] corpus. For segmentation, the accuracy of JTAG is

[4] JUMAN Version 3.4.
http://www-nagao.kuee.kyoto-u.ac.jp/index-e.html

[5] CHASEN Version 1.5.1.
http://cactus.aist-nara.ac.jp/lab/nlt/chasen.html

[6] The sentences do not include Arabic numerals because JUMAN and CHASEN do not assign phonological representation to them.

[7] Japan Electronic Dictionary Research Institute.
http://www.iijnet.or.jp/edr/

| | JTAG | JUMAN | CHASEN |
|---|---|---|---|
| Conversion Ratio | 88.5% | 71.7% | 72.3% |
| Processing Time | 86sec | 576sec | 335sec |

**Table III: Correct phonological representation per sentence. Average 38 characters in one sentence. Sun Ultra-1 170Mhz.**

the same as that of JUMAN. Table II shows that JTAG assigns the correct phonological representations to unsegmented Japanese sentences more precisely than do the other systems.

Table III shows the ratio of sentences that are converted to the correct phonological representation where segmentation errors are ignored. 80,000 sentences[8] (3,038,713 characters, no Arabic numerals) were used in the EDR corpus. The average number of characters in one sentence is 38. JTAG converts 88.5% of sentences correctly. The ratio is much higher than that of the other systems.

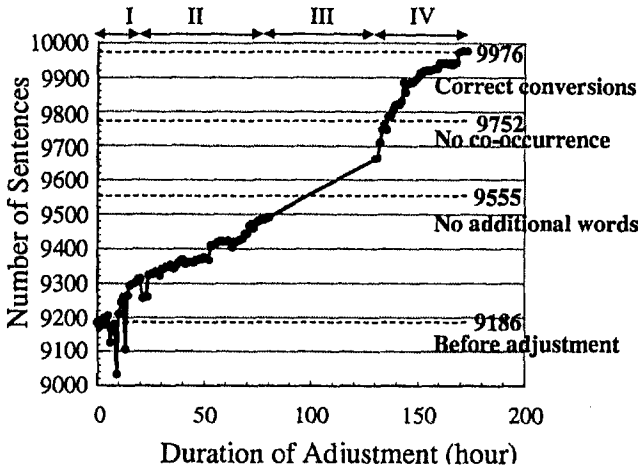
Table III also shows the processing time of each system. JTAG analyzes Japanese text more than do four times faster than the other taggers. The simplicity of the JTAG selection algorithm contributes to the fast processing speed.

## 3.2 Adjustment Process

To show the adjustablity of JTAG, we tuned it for a specific set of 10,000 sentences[9]. The average number of words in a sentence is 21. Graph 1 shows the transition of the number of sentences converted correctly to their phonological representation. We finished the adjustment when the system could no longer be tuned in the framework of JTAG. The last accuracy rating (99.8% per sentence) shows the maximum ability of JTAG.

The feature of each phase of the adjustment is described below.

**Phase I**. In this phase, the grammar of JTAG was changed. New attribute values were introduced and the costs of connection rules were changed.




**Graph 1: Transition of the number of sentences correctly converted to phonological representation.**

These adjustments caused large occurrences of degradation in our tagger.

**Phase II**. The grammar was almost fixed. One of the authors added unregistered words to the dictionaries, changed the costs of registered words, and supplied the information of the co-occurrence of words. The changes in the costs of words caused a small degree of degradation.

**Phase III**. In this phase, all unrecognized words were registered together. The unrecognized words were extracted automatically and checked manually. The time taken for this phase is the duration of the checking.

**Phase IV**. Mainly, co-occurrence information was supplied. This phase caused some degradation, but these instances were very small.

Graph 1 shows that JTAG converts 91.9% of open sentences to the correct phonological representation, and 99.8% of closed sentences. Without the co-occurrence information, the ratio is 97.5%. Therefore, the co-occurrence information corrects 2.3% of the sentences. Without new registered words, the ratio is 95.6%, so unrecognized words caused an error in 4.2% of the

| | Sentences | Errors |
|---|---|---|
| Unrecognized Words | 4.2% | 52% |
| Co-occurrence | 2.3% | 28% |
| Others | 1.6% | 20% |
| Total | 8.1% | 100% |

**Table IV: Causes of errors.**

[8] In the EDR corpus, 2.3% of sentences have errors and 1.5% of sentences have phonological representation inconsistencies. In this case, the sentences are not revised.

[9] 311,330 characters without Arabic numerals. Average 31 characters per sentence. In this case, we fixed all errors of the sentences and the inconsistency of their phonological representation.

sentences. Table IV shows the percentages of the causes.

## Conclusion

We developed a Japanese morphological analyzer that analyzes unsegmented Japanese sentences more precisely than other popular analyzers. Our system uses the co-occurrence of words to select the correct sequence of words. The efficiency of the co-occurrence information was shown through experimental results. The precision of our current tagger is 98.7% and the recall is 99.1%. The accuracy of the tagger can be expected to increase because the risk of degradation is small when using the co-occurrence information.

## References


Yoshimura K, Hitaka T. and Yoshida S. (1983) *Morphological Analysis of Non-marked-off Japanese Sentences by the Least BUNSETSU's Number Method.* Trans. IPSJ, Vol.24, No.1, pp.40-46. (in Japanese)

Miyazaki M. and Ooyama Y. (1986) *Linguistic Method for a Japanese Text to Speech System.* Trans. IPSJ, Vol.27, No.11, pp.1053-1059. (in Japanese)

Hisamitsu T. and Nitta Y. (1990) *Morphological Analysis by Minimum Connective-Cost Method.* SIGNLC 90-8, IEICE, pp.17-24. (in Japanese)

Brill E. (1992) *A simple rule-based part of speech tagger.* Procs. Of 3rd Conference on Applied Naural Language Processing, ACL.

Maruyama M. and Ogino S. (1994) *Japanese Morphological Analysis Based on Regular Grammar.* Trans. IPSJ, Vol.35, No.7, pp.1293-1299. (in Japanese)

Nagata M. (1994) *A Stochastic Japanese Morphological Analyzer Using a Forward-DP Backward-A* N-Best Search Algorithm.* Computational Linguistics, COLING, pp.201-207.

Fuchi T. and Yonezawa M. (1995) *A Morpheme Grammar for Japanese Morphological Analyzers.* Journal of Natural Language Processing, The Association for Natural Language Processing, Vol2, No.4, pp.37-65.

Pierre C. and Tapanainen P. (1995) *Tagging French – comparing a statical and a constraint-based method.* Procs. Of 7th Conference of the European Chapter of the ACL, ACL, pp.149-156.

Takeuchi K. and Matsumoto Y. (1995) *HMM Parameter Learning for Japanese Morphological Analyzer.* Procs. Of 10th Pacific Asia Conference Language, Information and Computation, pp.163-172.

Voutilainen A. (1995) *A syntax-based part of speech analyser.* Procs. Of 7th Conference of the European Chapter of the Association for Computational Linguistics, ACL, pp.157-164.

Matsuoka K., Takeishi E. and Asano H. (1996) *Natural Language Processing in a Japanese Text-To-Speech System for Written-style Texts.* Procs. Of 3rd IEEE Workshop On Interactive Voice Technology For Telecommunications Applications, IEEE, pp.33-36.

Samuelsson C. and Voutilainen A. (1997) *Comparing a Linguistic and a Stochastic Tagger.* Procs. Of 35th Annual Meeting of the Association for Computational Linguistics, ACL.


## Appendix

```
ELEMENT selection(SET sequences) {
  ELEMENT selected;
  int best_total_connective_cost = MAX_INT;
  int best_number_of_cooc = -1;
  int best_total_word_cost = -1;
  int best_number_of_2character_word = -1;
  foreach s (sequences) {
    s.total_connective_cost
      = sum_of_connective_cost(s);
    if (best_total_connective_cost
          > s.total_connective_cost) {
      best_total_connective_cost
          = s.total_connective_cost;
      selected = s; }}
  foreach s (sequences) {
    if (s.total_connective_cost
          - best_total_connective_cost
          > PRUNE_RANGE) {
      sequcences.delete(s); }}
  foreach s (sequences) {
    s.number_of_cooc
      = count_cooccurence_of_words(s);
    if (best_number_of_cooc
          < s.number_of_cooc) {
      best_number_of_cooc
          = s.number_of_cooc;
      selected = s; }}
  foreach s (sequences) {
    if (s.number_of_cooc
          < best_number_of_cooc) {
      sequcences.delete(s); }}
  foreach s (sequences) {
    s.total_word_cost
      = sum_of_word_cost(s);
    if (best_total_word_cost
          > s.total_word_cost) {
      best_total_word_cost
          = s.total_word_cost;
      selected = s; }}
  foreach s (sequences) {
    if (s.total_word_cost
          > best_total_word_cost) {
      sequcences.delete(s); }}
  foreach s (sequences) {
    s.number_of_2character_word
      = count_2character_word(s);
    if (best_number_of_2character_word
          < s.number_of_2character_word) {
      best_number_of_2character_word
          = s.number_of_2character_word;
      selected = s; }}
  return selected;
}
```