Use of Mutual Information Based Character Clusters in Dictionary-less Morphological Analysis of Japanese

Hideki Kashioka, Yasuhiro Kawata, Yumiko Kinjo, Andrew Finch and **Ezra W. Black**
{kashioka, ykawata, kinjo, finch, black}@itl.atr.co.jp
ATR Interpreting Telecommunications Reserach Laboratories

Abstract

For languages whose character set is very large and whose orthography does not require spacing between words, such as Japanese, tokenizing and part-of-speech tagging are often the difficult parts of any morphological analysis. For practical systems to tackle this problem, uncontrolled heuristics are primarily used. The use of information on character sorts, however, mitigates this difficulty. This paper presents our method of incorporating character clustering based on mutual information into Decision-Tree Dictionary-less morphological analysis. By using natural classes, we have confirmed that our morphological analyzer has been significantly improved in both tokenizing and tagging Japanese text.

1 Introduction

Recent papers have reported cases of successful part-of-speech tagging with statistical language modeling techniques (Church 1988; Cutting et. al. 1992; Charniak et. al. 1993; Brill 1994; Nagata 1994; Yamamoto 1996). Morphological analysis on Japanese, however, is more complex because, unlike European languages, no spaces are inserted between words. In fact, even native Japanese speakers place word boundaries inconsistently. Consequently, individual researchers have been adopting different word boundaries and tag sets based on their own theory–internal justifications.

For a practical system to utilize the different word boundaries and tag sets according to the demands of an application, it is necessary to coordinate the dictionary used, tag sets, and numerous other parameters. Unfortunately, such a task is costly. Furthermore, it is difficult to maintain the accuracy needed to regulate the word boundaries. Also, depending on the purpose, new technical terminology may have to be collected, the dictionary has to be coordinated, but the problem of unknown words would still remain.

The above problems will arise so long as a dictionary continue to play a principal role. In analyzing Japanese, a Decision-Tree approach with no need for a dictionary (Kashioka, et. al. 1997) has led us to employ, among other parameters, mutual information (MI) bits of individual characters derived from large hierarchically clustered sets of characters in the corpus.

This paper therefore proposes a type of Decision-Tree morphological analysis using the MI of characters but with no need for a dictionary. Next the paper describes the use of information on character sorts in morphological analysis involving the Japanese language, how knowing the sort of each character is useful when tokenizing a string of characters into a string of words and when assigning parts-of-speech to them, and our method of clustering characters based on MI bits. Then, it proposes a type of Decision-Tree analysis where the notion of MI-based character and word clustering is incorporated. Finally, we move on to an experimental report and discussions.

2 Use of Information on Characters

Many languages in the world do not insert a space between words in the written text. Japanese is one of them. Moreover, the number of characters involved in Japanese is very large. [1]

[1] Unlike English being basically written in a 26-character alphabet, the domain of possible characters appearing in an average Japanese text is a set involving tens of thousands of characters.

2.1 Character Sort

There are three clearly identifiable character sorts in Japanese: [2]

Kanji are Chinese characters adopted for historical reasons and deeply rooted in Japanese. Each character carries a semantic sense.

Hiragana are basic Japanese phonograms representing syllables. About fifty of them constitute the syllabary.

Katakana are characters corresponding to hiragana, but their use is restricted mainly to foreign loan words.

Each character sort has a limited number of elements, except for Kanji whose exhaustive list is hard to obtain.

Identifying each character sort in a sentence would help in predicting the word boundaries and subsequently in assigning the parts-of-speech. For example, between characters of different sorts, word boundaries are highly likely. Accordingly, in formalizing heuristics, character sorts must be assumed.

2.2 Character Cluster

Apart from the distinctions mentioned above, are there things such as natural classes with respect to the distribution of characters in a certain set of sentences (therefore, the classes are empirically learnable)? If there are, how can we obtain such knowledge?

It seems that only a certain group of characters tends to occur in a certain restricted context. For example, in Japanese, there are many numerical classifier expressions attached immediately after numericals. [3] If such is the case, these classifiers can be clustered in terms of their distributions with respect to a presumably natural class called numericals. Supposing one of a certain group of characters often occurs as a neighbor to one of the other groups of characters, and supposing characters are clustered and organized in a hierarchical fashion, then it is possible to refer to such groupings by pointing out a certain node in the structure. Having a way of organizing classes of characters is clearly an advantage in describing facts in Japanese. The next section presents such a method.

3 Mutual Information-Based Character Clustering

One idea is to sort words out in terms of neighboring contexts. Accordingly research has been carried out on n-gram models of word clustering (Brown et. al. 1992) to obtain hierarchical clusters of words by classifying words in such a way so as to minimizes the reduction of MI.

This idea is general in the clustering of any kind of list of items into hierarchical classes. [4] We therefore have adopted this approach not only to compute word classes but also to compute character clusterings in Japanese.

The basic algorithm for clustering items based on the amount of MI is as follows: [5]

1) Assign a singleton class to every item in the set.
2) Choose two appropriate classes to create a new class which subsumes them.
3) Repeat 2) until the additional new items include all of the items in the set.

With this method, we conducted an experimental clustering over the ATR travel conversation corpus. [6] As a result, all of the characters in the corpus were hierarchically clustered according to their distributions.

Example: A partial character clustering

```
-+---------- 得 0000000110111
 +-+-+-+---- 列 0000000111000000
   | | +-+-  駐 00000001110000010
   | |   +-  汽 00000001110000011
   | +-----  停 000000011100001
   +-------  包 00000001110001000
```

Each node represents a subset of all of the different characters found in the training data. We represent tree structured clusters with bit strings, so that we may specify any node in the structure by using a bit substring.

[2] Other sorts found in ordinary text are Arabic numerics, punctuations, other symbols, etc.

[3] For example, "3 冊 (san-satsu)" for bound objects "3 copies of", "2 枚 (ni-mai)" for flat objects "2 pieces/sheets of".

[4] Brown, et. al. (1992) for details.

[5] This algorithm, however, is too costly because the amount of computation exponentially increases depending on the number of items. For practical processing, the basic procedure is carried out over a certain limited number of items, while a new item is supplied to the processing set each time clustering is done.

[6] 80,000 sentences, with a total number of 1,585,009 characters and 1,831 different characters.

Numerous significant clusters are found among them. [7] They are all natural classes computed based on the events in the training set.

4 Decision-Tree Morphological Analysis

The Decision-Tree model consists of a set of questions structured into a dendrogram with a probability distribution associated with each leaf of the tree. In general, a decision-tree is a complex of n-ary branching trees in which questions are associated with each parent node, and a choice or class is associated with each child node. [8] We represent answers to questions as bits.

Among other advantages to using decision-trees, it is important to note that they are able to assign integrated costs for classification by all types of questions at different feature levels provided each feature has a different cost.

4.1 Model

Let us assume that an input sentence $C = c_1\ c_2\ ...\ c_n$ denotes a sequence of n characters that constitute words $W = w_1\ w_2\ ...\ w_m$, where each word w_i is assigned a tag t_i ($T = t_1\ t_2\ ...\ t_m$).

The morphological analysis task can be formally defined as finding a set of word segmentations and part-of-speech assignments that maximizes the joint probability of the word sequence and tag sequence $P(W,T|C)$.

The joint probability $P(W,T|C)$ is calculated by the following formulae:

$$P(W,T|C) = \prod_{i=1}^{M} P(w_i,t_i|w_1,...,w_{i-1},t_1,...,t_{i-1},C)$$

$$P(w_i,t_i|w_1,...,w_{i-1},t_1,...,t_{i-1},C) = P(w_i|w_1,...,w_{i-1},t_1,...,t_{i-1},C)^{9} * P(t_i|w_1,...,w_i,t_1,...,t_{i-1},C)^{10}$$

The Word Model decision-tree is used as the word tokenizer. While finding word boundaries, we use two different labels: **Word+** and **Word–**. In the training data, we label **Word+** to a complete word string, and **Word–** to every substring of a relevant word since these substrings are not in fact a word in the current context. [11] The probability of a word estimates the associated distributions of leaves with a word decision-tree.

We use the Tagging Model decision-tree as our part-of-speech tagger. For an input sentence C, let us consider the character sequence from c_1 to c_{p-1} (assigned $w_1\ w_2\ ...\ w_{k-1}$) and the following character sequence from p to $p+l$ to be the word w_k; also, the word w_k is assumed to be assigned the tag t_k.

We approximate the probability of the word w_k assigned with tag t_k as follows: $P(t_k) = p(t_i|w_1,...,w_k,t_1,...,t_{k-1},C)$. This probability estimates the associated distributions of leaves with a part-of-speech tag decision-tree.

4.2 Growing Decision-Trees

Growing a decision-tree requires two steps: selecting a question to ask at each node; and determining the probability distribution for each leaf from the distribution of events in the training set. At each node, we choose from among all possible questions, the question that maximizes the reduction in entropy.

The two steps are repeated until the following conditions are no longer satisfied:

- The number of leaf node events exceeds the constant number.
- The reduction in entropy is more than the threshold.

Consequently, the list of questions is optimally structured in such a way that, when the data flows in the decision-tree, at each decision point, the most efficient question is asked.

Provided a set of training sentences with word boundaries in which each word is assigned with a part-of-speech tag, we have a) the necessary structured character clusters, and b) the necessary structured word clusters; [12] both of them are based on the n-gram language model.

[7] For example, katakana, numerical classifiers, numerics, postpositional case particles, and prefixes of demonstrative pronouns.

[8] The work described here employs only binary decision-trees. Multiple alternative questions are represented in more than two yes/no questions. The main reason for this is the computational efficiency. Allowing questions to have more answers complicates the decision-tree growth algorithm.

[9] We call this the "Word Model".

[10] We call this the "Tagging Model".

[11] For instance, for the word "mo-shi-mo-shi" (hello), "mo-shi-mo-shi" is labeled **Word+**, and "mo-shi-mo", "mo-shi", "mo" are all labeled **Word–**. Note that "mo-shi" or "mo-shi-mo" may be real words in other contexts, e.g., "mo-shi/wa-ta-shi/ga . . . (If I do . . .)".

[12] Here, a word token is based only on a word string, not on a word string tagged with a part-of-speech.

We also have c) the necessary decision-trees for word-splitting and part-of-speech tagging, each of which contains a set of questions about events. We have considered the following points in making decision-tree questions.

1) **MI character bits**
We define self-organizing character classes represented by binary trees, each of whose nodes are significant in the n-gram language model. We can ask which node a character is dominated by.
2) **MI word bits**
Likewise, MI word bits (Brown et. al. 1992) are also available so that we may ask which node a word is dominated by.
3) **Questions about the target word**
These questions mostly relate to the morphology of a word (e.g., Is it ending in '-shi-i' (an adjective ending)? Does it start with 'do-'?).
4) **Questions about the context**
Many of these questions concern continuous part-of-speech tags (e.g., Is the previous word an adjective?). However, the questions may concern information at different remote locations in a sentence (e.g., Is the initial word in the sentence a noun?).

These questions can be combined in order to form questions of greater complexity.

5 Analysis with Decision-Trees

Our proposed morphological analyzer processes each character in a string from left to right. Candidates for a word are examined, and a tag candidate is assigned to each word. When each candidate for a word is checked, it is given a probability by the word model decision-tree. We can either exhaustively enumerate and score all of the cases or use a stack decoder algorithm (Jelinek 1969; Paul 1991) to search through the most probable candidates.

The fact that we do not use a dictionary, [13] is one of the great advantages. By using a dictionary, a morphological analyzer has to deal with unknown words and unknown tags, [14] and is also fooled by many words sharing common substrings. In practical contexts, the system refers to the dictionary by using heuristic rules to find the more likely word boundaries, e.g., the minimum number of words, or the maximum word length available at the minimum cost. If the system could learn how to find word boundaries without a dictionary, then there would be no need for such an extra device or process.

Table 1: Travel Conversation

Training	A (%)	B (%)
1,000+MIChr	80.67	69.93
-MIChr	70.03	62.24
2,000+MIChr	86.61	76.43
-MIChr	69.65	63.36
3,000+MIChr	88.60	79.33
-MIChr	71.97	66.47
4,000+MIChr	88.26	80.11
-MIChr	72.55	67.24
5,000+MIChr	89.42	81.94
-MIChr	72.41	67.72

Training: number of sentences
with/without Character Clustering
A: Correct word/system output words
B: Correct tags/system output words

6 Experimental Results

We tested our morphological analyzer with two different corpora: a) ATR-travel, which is a task oriented dialogue in a travel context, and b) EDR Corpus, (EDR 1996) which consists of rather general written text.

For each experiment, we used the character clustering based on MI. Each question for the decision-trees was prepared separately, with or without questions concerning the character clusters. Evaluations were made with respect to the original tagged corpora, from which both the training and test sentences were taken.

The analyzer was trained for an incrementally enlarged set of training data using or not using character clustering. [15] Table 1 shows results obtained from training sets of ATR-travel. The upper figures in each box indicate the results when using the character clusters, and the lower without using them. The actual test set of 4,147 sentences (55,544 words) was taken from

[13] Here, a dictionary is a listing of words attached to part-of-speech tags.

[14] Words that are not found in the dictionary and necessary tags that are not assigned in the dictionary.

[15] Another 2,231 sentences (28,933 words) in the same domain are used for the smoothing.

Table 2: General Written Text

Training	A (%)	B (%)
3,000+MIChr	83.80	78.19
-MIChr	77.56	72.49
5,000+MIChr	85.50	80.42
-MIChr	78.68	73.84
7,000+MIChr	85.97	81.66
-MIChr	79.32	75.30
9,000+MIChr	86.08	81.20
-MIChr	78.59	74.05
10,000+MIChr	86.22	81.39
-MIChr	78.94	74.41

the same domain.

The MI-word clusters were constructed according to the domain of the training set. The tag set consisted of 209 part-of-speech tags. [16] For the word model decision-tree, three of 69 questions concerned the character clusters and three of 63 the tagging model. Their presence or absence was the deciding parameter.

The analyzer was also trained for the EDR Corpus. The same character clusters as with the conversational corpus were used. A tag set in the corpus consisted of 15 parts-of-speech. For the word model, 45 questions were prepared; 18 for the Tagging model. Just a couple of them were involved in the character clusters. The results are shown in Table 2.

7 Conclusion and Discussion

Both results show that the use of character clusters significantly improves both tokenizing and tagging at every stage of the training. Considering the results, our model with MI characters is useful for assigning parts of speech as well as for finding word boundaries, and overcoming the unknown word problem.

The consistent experimental results obtained from the training data with different word boundaries and different tag sets in the Japanese text, suggests the method is generally applicable to various different sets of corpora constructed for different purposes. We believe that with the appropriate number of adequate questions, the method is transferable to other languages that have word boundaries not indicated in the text.

In conclusion, we should note that our method, which does not require a dictionary, has been significantly improved by the character cluster information provided.

Our plans for further research include investigating the correlation between accuracy and the training data size, the number of questions as well as exploring methods for factoring information from a "dictionary" into our model. Along these lines, a fruitful approach may be to explore methods of coordinating probabilistic decision-trees to obtain a higher accuracy.

[16] These include common noun, verb, post-position, auxiliary verb, adjective, adverb, etc. The purpose of this tag set is to perform machine translation from Japanese to English, German and Korean.

References

Brill, E. (1994) "Some Advances in Transformation-Based Part of Speech Tagging," AAAI-94, pp. 722–727.

Brown, P., Della Pietra, V., de Souza, P., Lai, J., and Mercer, R. (1992) "Class-based n-gram models of natural language," *Computational Linguistics*, Vol. 18, No. 4, pp. 467–479.

Cutting, D., Kupiec, J., Pedersen, J., and Sibun, P. (1992) "A Practical Part-of-Speech Tagger," ANLP-92, pp. 133–140.

Charniak, E., Hendrickson, C., Jacobson, N., and Perkowits, M. (1993) "Equations for Part-of-Speech Tagging," AAAI-93, pp. 784–789.

Church, K. (1988) "A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text," *Proceedings of the 2nd Conference on Applied Natural Language Processing*, Austin-Marriott at the Capitol, Austin, Texas, USA, 1988, pp. 136-143.

EDR (1996) *EDR Electronic Dictionary Version 1.5 Technical Guide.* EDR TR2-007.

Jelinek, F. (1969) "A fast sequential decoding algorithm using a stack," *IBM Journal of Research and Development*, Vol. 13, pp. 675–685.

Kashioka, H., Black, E., and Eubank, S. (1997) "Decision-Tree Morphological Analysis without a Dictionary for Japanese," *Proceedings of NLPRS 97*, pp. 541-544.

Nagata, M. (1994) "A Stochastic Japanese Morphological Analyzer Using a Forward-DP Backward-A* N-Best Search Algorithm," *Proceedings of COLING-94*, pp. 201–207.

Paul, D. (1991) "Algorithms for an optimal a^* search and linearizing the search in the stack decoder," *Proceedings, ICASSP 91*, pp. 693-696.

Yamamoto, M. (1996) "A Re-estimation Method for Stochastic Language Modeling from Ambiguous Observations," *WVLC-4*, pp. 155–167.