Language, Information and Computation (PACLIC12), 18-20 Feb, 1998, 230-235

Automatic Bunsetsu Segmentation of Japanese Sentences Using a Classification Tree

Yujie Zhang and Kazuhiko Ozeki**
The University of Electro-Communications

Bunsetsu, which is comprised of a content word followed by, possibly 0, function words, is a convenient unit for dependency structure analysis of Japanese. There are, however, no spaces indicating bunsetsu boundaries in the orthographic writing of Japanese. Thus a sentence must be segmented into bunsetsu's by some means prior to dependency structure analysis. Conventionally, such segmentation has been performed by using some kind of hand-crafted rules. This paper describes a novel segmentation method using a classification tree, by which knowledge about bunsetsu boundaries is automatically acquired from a labeled corpus. The method enables quick and easy adaptation to a new task domain, and also to a new system of morpheme categorization without the need of changing the algorithm. Effectiveness of this method is shown through experiments on an ATR corpus and an EDR corpus.

1. INTRODUCTION

Bunsetsu, which is comprised of a content word with or without being followed by a string of function words, is a convenient unit for dependency structure analysis of Japanese. There are, however, no spaces indicating bunsetsu boundaries in the orthographic writing of Japanese. Thus a sentence must be segmented into bunsetsu's prior to dependency structure analysis. According to the elementary definition of bunsetsu [Nagao, ed. (1984)], such segmentation might look simple. There are, in reality, many factors that complicate the problem. For example, a prefix and/or a suffix can be attached to a content word, and Chinese characters can be concatenated to form a compound word. Some nouns and verbs have functions different from their original ones. Also, there are many idiomatic usages of morpheme concatenations. All these matters cause difficulties in detecting bunsetsu boundaries. Moreover, there is no system of morpheme categorization in Japanese that has received a general consensus. This situation gives rise to another obstacle to establishing a standard method of bunsetsu segmentation.

There have been two major approaches to the bunsetsu segmentation problem: one based on an automaton [Fujio *et al.* (1997)], or bunsetsu patterns [Kurohashi (1997)] representing a definition of bunsetsu, and the other based on a set of hand-crafted rules [Suzuki (1996)]. In the former approach, one has to give a definition of bunsetsu manually. The latter involves handiwork in getting knowledge about bunsetsu boundaries. Thus both approaches have problems in keeping consistency, coverage and optimality of manually obtained knowledge. When the task domain and/or the system of morpheme categorization are changed, one has to repeat the whole manual process to get new knowledge, which is rather laborious.

This paper proposes a method of bunsetsu segmentation using a classification tree [Breiman *et al.* (1984)], [Quinlan (1993)], by which knowledge about bunsetsu boundaries is automatically acquired from a corpus. It enables quick adaptation to a new task domain, and also to a new system of morpheme categorization without the need of changing the algorithm. Effectiveness of the method is shown through experiments on an ATR corpus and an EDR corpus.

2. CLASSIFICATION TREE

A classification tree is a binary tree that classifies objects into classes [Breiman *et al.* (1984)], [Quinlan (1993)]. Through an automatic generation of a classification tree, one can rapidly acquire underlying regularities in a large amount of data, which are difficult or even impossible for a human to capture by

*Department of Computer Science and Information Mathematics, The University of Electro-Communications
1-5-1 Chofugaoka, Chofu, Tokyo, 182-8585 Japan. Email: zhang@achilleus.cs.uec.ac.jp, ozeki@cs.uec.ac.jp

intuition. This technique has been well studied in such fields as pattern recognition and machine learning [Safavian *et al.* (1991)]. A number of applications to natural language processing and speech processing have also been reported [Kuhn *et al.* (1995)], [Wang *et al.* (1992)], [Ostendorf *et al.* (1993)].

The process of classification by a tree is shown in Fig.1. Associated with each non-terminal node of a classification tree is a *test*. If an object passes a test, it reaches "yes" child-node; otherwise "no" child-node. The process is repeated until the object reaches some terminal node or *leaf*, which has been assigned to a class label. Only two classes, c_1 and c_2, will be considered here.

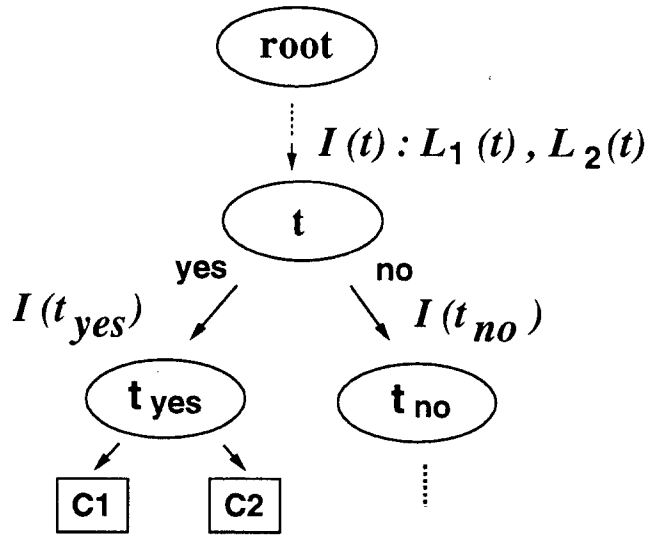

Fig.1 Classification tree for two classes c_1 and c_2.

In order to grow a classification tree from the root, objects with class labels, or *training* objects, are necessary. Also a finite set of tests must be prepared. Suppose that a tree has been grown to some size. It consists of three kinds of nodes: non-terminal nodes, leaves, and active nodes. An active node is a tentative terminal node that will be turned into a non-terminal node or a leaf afterward. Let t be an active node, and $L(t)$ the number of training objects that reach t, among which the number of objects belonging to c_i is denoted as $L_i(t)$ $(i = 1, 2)$. Then the impurity of t, *Gini* index [Breiman *et al.* (1984)], is defined as

$$I(t) = \frac{L_1(t)}{L(t)} \cdot \frac{L_2(t)}{L(t)}.$$

If the impurity is lower than a prescribed threshold, then t is decided to be a leaf. Otherwise all the tests are tried exhaustively. By means of a test, the training objects that reach t are divided into those that reach the "yes" child-node t_{yes} and those that reach the "no" child-node t_{no}. Let $L(t_x)$ denote the number of training objects that reach t_x, where x equals "yes" or "no". Then the test that maximizes the reduction of impurity

$$\triangle I(t) = I(t) - \frac{L(t_{yes})}{L(t)} I(t_{yes}) - \frac{L(t_{no})}{L(t)} I(t_{no})$$

is selected as the test associated with t, and new active nodes, t_{yes} and t_{no}, are appended under t. If there is no test that reduces the impurity of t, then t is decided to be a leaf. With the root as the initial active node, the above procedure is iterated until all the active nodes are turned into non-terminal nodes or leaves. A leaf t is assigned to class label c_i if the majority of training objects that reach t belong to class c_i $(i = 1, 2)$.

3. APPLICATION OF CLASSIFICATION TREES TO BUNSETSU SEGMENTATION

By morphological analysis, a sentence is segmented into morphemes. The attribute values of each morpheme such as the part of speech and the orthographic expression are also obtained. An object to be classified here is a pair of a morpheme sequence derived from a sentence and a boundary between

morphemes (a boundary in focus, henceforth): $(m_1 m_2 \cdots m_n,\ b_i)$, where m_k $(1 \leq k \leq n)$ is a morpheme labeled with its attribute values, and b_i the boundary between m_i and m_{i+1}. Therefore a sentence consisting of N morphemes yields $N-1$ objects. The purpose of classification is to decide whether the boundary in focus is a bunsetsu boundary in the morpheme sequence; this is a classification problem for two classes.

Since it is expected that morphemes near the boundary in focus are important, only two morphemes adjacent to the boundary are tested: one immediately on the left (left morpheme) and the other immediately on the right (right morpheme). Among the attributes of a morpheme, the part of speech is considered to be most important. In some cases, however, the part of speech alone does not provide enough information for bunsetsu boundary detection. Therefore the orthographic expression is employed as another test attribute for some range of morphemes, which are selected by a preliminary experiment. Also the wild card "$*$" is introduced as a symbol to match any attribute value. Let p_i be a part of speech, and $W(p_i)$ the set of orthographic expressions, including "$*$", of morphemes that belong to p_i and are selected by the preliminary experiment. Then the set of the pairs of p_i and its orthographic expressions is denoted as $\{p_i\} \times W(p_i)$. Let S be the set of all such pairs plus $(*, *)$:

$$S = \sum_i (\{p_i\} \times W(p_i)) \cup \{(*, *)\}.$$

Then the set of all the tests is given by $S \times S$ in the present work. A test takes the form $< (p_1, e_1)(p_2, *) >$, for example. An object will pass this test if the part of speech of the left morpheme equals p_1, its orthographic expression equals e_1, and the part of speech of the right morpheme equals p_2. A similar technique has been applied to classification of intonational phrase boundaries [Wang *et al.* (1992)], though the purpose and the attributes used in their work are quite different from ours.

In the process of growing a classification tree, an active node t is decided to be a leaf if the condition $I(t) < T$ is satisfied for a prescribed threshold T, or if there is no test that reduces the impurity of t. In this work the value of T was set at 0.1. This condition implies that more than 90% of the training objects that reach t belong to the same class.

4. EXPERIMENTAL RESULTS

By the procedure described above, classification trees for the bunsetsu segmentation task were generated by using the training objects, and then the results were evaluated. In order to see the influence of different morpheme categorization systems and different sentence materials on the results, two corpora, an ATR corpus and an EDR corpus, were used in the experiments.

4.1 Experiment on ATR Corpus

The ATR corpus contains 503 sentences taken from newspapers, magazines, and etc. The sentences are segmented into morphemes, and labeled with the part of speech and the bunsetsu boundary [Abe *et al.* (1990)]. In this experiment 6994 objects were extracted from the corpus, of which 2000 were used for training, and the remaining 4994 for evaluation. The attribute used for test here was the part of speech only; the orthographic expression was not used. There are 25 parts of speech in the ATR corpus.

Fig.2 illustrates a part of the generated tree near the root. The total number of nodes was 69. There was a tendency that the tests related to morphemes with higher frequencies appeared in the nodes closer to the root. Thus it can be said that an efficient order of tests was realized automatically.

Table 1 shows the result of evaluation. The symbol Y signifies that the boundary in focus is a bunsetsu boundary, and N signifies that is not. The arrow $\rightarrow$ denotes the classification operation by the tree. So, $Y \rightarrow Y$ means that an object labeled as a bunsetsu boundary in the corpus is classified as a bunsetsu boundary by the tree. The expression $|Y \rightarrow Y|$ denotes the number of such cases.

About 76% of the errors were of the type $Y \rightarrow N$. It was found that most of the errors of this type came from concatenations of two common nouns. The generated tree decided a morpheme boundary between common nouns not to be a bunsetsu boundary. However, many of the first common nouns in such concatenations were in fact those that functioned as adverbs, and this kind of objects were labeled as bunsetsu boundaries in the corpus. Thus, the coarseness of sub-categorization of noun was a part of the causes of the errors.

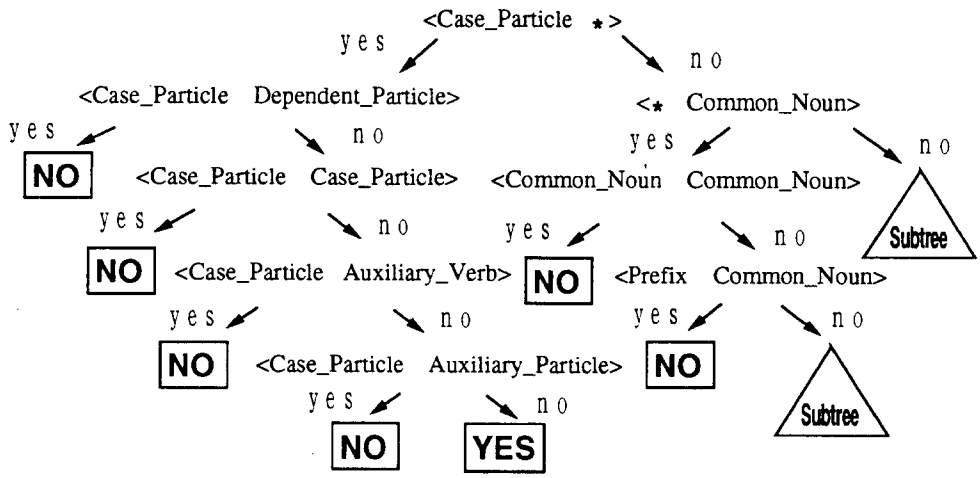

Fig.2 Part of the tree generated on the ATR corpus.

Table 1. Evaluation result for the ATR corpus.

No. of objects	$\|Y \rightarrow Y\|$	$\|N \rightarrow N\|$	$\|Y \rightarrow N\|$	$\|N \rightarrow Y\|$	% Accuracy
4994	2015	2893	65	21	98.3

4.2 Experiment on EDR Corpus

In the EDR corpus [EDR (1996)], there are no labels indicating bunsetsu boundaries. Instead, it has detailed information about the syntactic structure of sentences. By utilizing this information and the definition of bunsetsu [Nagao, ed. (1984)], 400 sentences were labeled with the bunsetsu boundary. Then 6984 objects for training were extracted from randomly selected 200 sentences, and 7110 objects for evaluation from the rest.

The sub-categorization of noun in the EDR corpus seemed too coarse for the present purpose. Therefore it was augmented by using the semantic identifier, which was common to the corpus and the dictionary. The resulting number of parts of speech was 19. In the EDR corpus, sub-categorization of particle is coarser than that in the ATR corpus. Moreover, there is no such category as formal verb, which is employed as a category in the ATR corpus. Therefore, the test of the form <(particle, *),(verb,*)> has little power to distinguish between a bunsetsu boundary and a non bunsetsu boundary; the part of speech information alone, especially for particles and verbs, is not enough for bunsetsu segmentation. For that reason, the orthographic expression was also used as an attribute for test together with the part of speech.

A classification tree with 175 nodes was generated in this case. It was observed that the tree generated on the EDR corpus (EDR tree, henceforth) acquired new segmentation rules that were not acquired in the tree generated on the ATR corpus (ATR tree, henceforth):

1. The boundary between a temporal noun and a common noun was decided to be a bunsetsu boundary, while it was not by the ATR tree. (Because the ATR corpus has no such category as temporal noun, it makes no distinction between a temporal noun and a common noun.)

2. The boundary between an auxiliary verb and a common noun was decided to be a bunsetsu boundary, and a boundary between an auxiliary verb and a formal noun was not, while neither was decided to be a bunsetsu boundary by the ATR tree. (Because the ATR corpus has no such category as formal noun, it makes no distinction between a common noun and a formal noun.)

3. The boundary between a particle and a common noun was decided to be a bunsetsu boundary, and the boundary between a particle and a formal noun was not, while both were decided to be bunsetsu boundaries by the ATR tree. (The cause is the same as above.)

4. The boundary between two common nouns was decided to be a bunsetsu boundary, and the boundary between two proper nouns was not, while neither was decided to be a bunsetsu boundary by the ATR tree. (The ATR corpus does not contain enough amount of data for extracting such a segmentation rule.)

5. The EDR tree acquired 12 rules related to symbols, while the ATR tree acquired no such rules. (The ATR corpus has no occurence of symbols.)

Thus, the classification tree extracted the new segmentation rules exploiting the sub-categorization of noun in the EDR corpus. It is observed that in this way a classification tree can adapt to a new system of morpheme categorization.

Table 2. Evaluation result for the EDR corpus.

No. of objects	$\|Y \rightarrow Y\|$	$\|N \rightarrow N\|$	$\|Y \rightarrow N\|$	$\|N \rightarrow Y\|$	% Accuracy
7110	2502	4341	62	205	96.2

Table 2 shows the result of evaluation. The causes of the errors have been analyzed as follows.

- In this experiment, only two morphemes, one on the left and the other on the right of the boundary in focus, were tested. There were, however, some cases where testing more than two morphemes would improve the result.
- The set of orthographic expressions used as attribute values in the tests was incomplete. There were cases where adding some other orthographic expressions would yield better results.
- The training objects did not cover all the linguistic phenomena concerning bunsetsu boundaries. Thus sparseness of the training objects was a problem.
- Some errors obviously resulted from mislabeling in the corpus.

5. CONCLUSION

The results of this work are summarized as follows:

- A classification tree can acquire linguistic knowledge about bunsetsu boundaries automatically, given an appropriately labeled corpus.
- Using the criterion of maximum impurity reduction, it generates efficient rules that capture statistical and logical regularities concerning bunsetsu boundaries.
- It enalbes quick adaptation to a new task domain, and also to a new system of morpheme categorization without the need of changing the algorithm.

Our future work includes:

- Improvement of the control method for growing a classification tree by adjusting the threshold value T, or by pruning [Breiman *et al.* (1984)], [Gelfand *et al.* (1991)], [Quinlan (1993)], so that a better generalization can be attained.
- Pursuit of a better method for assigning leaves to class labels to enhance the reliability of decision at leaves.
- Automatic acquisition of morphemes whose orthographic expressions are effective in bunsetsu boundary detection.

REFERENCES

1. Abe, Masanobu; Sagisaka, Yoshinori; Umeda, Tetsuo; and Kuwabara, Hisao (1990). *Speech Database User's Manual.* ATR Interpreting Telephony Research Laboratories (in Japanese).

2. Breiman, Leo; Friedman, Jerome H.; Olshen, Richard A.; and Stone, Charles A. (1984). *Classification and Regression Trees.* Chapman and Hall.

3. EDR (Japan Electronic Dictionary Research Institute) (1996). *Specifications of EDR Electronic Dictionary Ver.1.5* (in Japanese).

4. Fujio, Masakazu; and Matsumoto, Yuji (1997). Statistical Japanese Dependency Structure Analysis Using an EDR Bracketed Corpus. *Proc. of Symposium on Applications of EDR Electronic Dictionary*:49-55 (in Japanese).

5. Gelfand, Saul B.; Ravishankar, C.S.; and Delp, Edward J. (1991). An Iterative Growing and Pruning Algorithm for Classification Tree Design. *IEEE Trans. PAMI* 13(2):163-174.

6. Kuhn, Roland; and De Mori, Renato (1995). The Application of Semantic Classification Trees to Natural Language Understanding. *IEEE Trans. PAMI* 17(5): 449-460.

7. Kurohashi, Sadao (1997). *Japanese Parsing System, KNP version 2.0 b3, User's Manual.* Faculty of Engineering, Kyoto University (in Japanese).

8. Nagao, Makoto, ed. (1984). *Japanese Language Information Processing.* The Institute of Electronics, Information and Communication Engineers (in Japanese).

9. Ostendorf, M.; Wightman, C.W.; and Veilleux, N.M. (1993). Parse Scoring with Prosodic Information: an Analysis/Synthesis Approach. *Computer Speech and Language* 7: 193-210.

10. Quinlan, J.Ross (1993). *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers.

11. Safavian, S.Rasoul; and Landgrebe, David (1991). A Survey of Decision Tree Classifier Methodology. *IEEE Trans. SMC* 21(3): 660-674.

12. Suzuki, Emiko (1996). Japanese Sentence Segmentation Algorithm Using Character Patterns Based on the Statistical Investigation. *IEICE Trans. on Information and Systems* J79-D-II(7):1236-1243 (in Japanese).

13. Wang, Michelle Q.; and Hirshberg, Julia (1992). Automatic Classification of Intonational Phrase Boundaries. *Computer Speech and Language* 6: 175-196.