Dependency Structure Analysis and Sentence Boundary Detection in Spontaneous Japanese

Kazuya Shitaoka[†] **Kiyotaka Uchimoto**[‡] **Tatsuya Kawahara**[†] **Hitoshi Isahara**[‡]

[†]School of Informatics,
Kyoto University
Yoshida-honmachi, Sakyo-ku,
Kyoto 606-8501, Japan,
{shitaoka,kawahara}@ar.media.kyoto-u.ac.jp

[‡]National Institute of Information
and Communications Technology
3-5 Hikari-dai, Seika-cho, Soraku-gun,
Kyoto 619-0289, Japan,
{uchimoto,isahara}@nict.go.jp

Abstract

This paper describes a project to detect dependencies between Japanese phrasal units called *bunsetsu*s, and sentence boundaries in a spontaneous speech corpus. In monologues, the biggest problem with dependency structure analysis is that sentence boundaries are ambiguous. In this paper, we propose two methods for improving the accuracy of sentence boundary detection in spontaneous Japanese speech: One is based on statistical machine translation using dependency information and the other is based on text chunking using SVM. An F-measure of 84.9 was achieved for the accuracy of sentence boundary detection by using the proposed methods. The accuracy of dependency structure analysis was also improved from 75.2% to 77.2% by using automatically detected sentence boundaries. The accuracy of dependency structure analysis and that of sentence boundary detection were also improved by interactively using both automatically detected dependency structures and sentence boundaries.

1 Introduction

The “Spontaneous Speech: Corpus and Processing Technology” project has been sponsoring the construction of a large spontaneous Japanese speech corpus, *Corpus of Spontaneous Japanese (CSJ)* (Maekawa et al., 2000). The CSJ is the biggest spontaneous speech corpus in the world, and it is a collection of monologues and dialogues, the majority being monologues such as academic presentations. The CSJ includes transcriptions of speeches as well as audio recordings. Approximately one tenth of the CSJ has been manually annotated with information about morphemes, sentence boundaries, dependency structures, discourse structures, and so on. The remaining nine tenths of the CSJ have been annotated semi-automatically. A future goal of the project is to extract sentence boundaries, dependency structures, and discourse structures from the remaining transcriptions. This paper focuses on methods for automatically detecting sentence boundaries and dependency structures in Japanese spoken text.

In many cases, Japanese dependency structures are defined in terms of the dependency relationships between Japanese phrasal units called *bunsetsu*s. To define dependency relationships between all *bunsetsu*s in spontaneous speech, we need to define not only the dependency structures in all sentences but also the inter-sentential relationships, or, discourse relationships, between the sentences, as dependency relationships between *bunsetsu*s. However, it is difficult to define and detect discourse relationships between sentences because of significant inconsistencies in human annotations of discourse structures, especially with regard to spontaneous speech. We also need to know intra-sentential dependency structures in order to use the results of dependency structure analysis for sentence compaction in automatic text summarization or case frame acquisition. Because it is difficult to define discourse relationships between sentences, depending on the actual application, it is usually enough to define and detect the dependency structure of each sentence. Therefore, the CSJ was annotated with intra-sentential dependency structures for sentences in the same way this is usually done for a written text corpus. However, there is a big difference between a written text corpus and a spontaneous speech corpus: In spontaneous speech, especially when it is long, sentence boundaries are often ambiguous. In the CSJ, therefore, sentence boundaries were defined based on clauses whose boundaries were automatically detected by using surface information (Maruyama et al., 2003), and they were detected manually (Takanashi et al., 2003). Our definition of sentence boundaries follows the definition used in the CSJ.

Almost all previous research on Japanese dependency structure analysis dealt with dependency structures in written text (Fujio and Matsumoto, 1998; Haruno et al., 1998; Uchimoto et al., 1999; Uchimoto et al., 2000; Kudo and Matsumoto, 2000). Although Matsubara and colleagues did investigate dependency structures in spontaneous speech (Matsubara et al., 2002), the target speech was dialogues where the utterances were short and sentence boundaries could be easily defined based on turn-taking data. In contrast, we investigated dependency structures in spontaneous and long speeches in the CSJ. The biggest problem in dependency structure analysis with spontaneous and long speeches is that sentence boundaries are ambiguous. Therefore, sentence boundaries should be detected before or during dependency structure analysis in order to obtain the dependency structure of each sentence.

In this paper, we first describe the problems with dependency structure analysis of spontaneous speech. Because the biggest problem is ambiguous sentence boundaries, we focus on sentence boundary detection and propose two methods for improving the accuracy of detection.

2 Dependency Structure Analysis and Sentence Boundary Detection in Spontaneous Japanese

First, let us briefly describe how dependency structures can be represented in a Japanese sentence. In Japanese sentences, word order is rather free, and subjects and objects are often omitted. In languages having such characteristics, the syntactic structure of a sentence is generally represented by the relationship between phrasal units, or *bunsetsu*s, based on a dependency grammar. Phrasal units, or *bunsetsu*s, are minimal linguistic units obtained by segmenting a sentence naturally in terms of semantics and phonetics. Each *bunsetsu* consists of one or more morphemes. For example, the sentence "彼はゆっくり歩いている (*kare-wa yukkuri aruite-iru*, He is walking slowly)" can be divided into three *bunsetsu*s, "彼は (*kare-wa*, he)", "ゆっくり (*yukkuri*, slowly)" and "歩いている (*aruite-iru*, is walking)". In this sentence, the first and second *bunsetsu*s depend on the third one.

There are many differences between written text and spontaneous speech, and there are problems peculiar to spontaneous speech in dependency structure analysis and sentence boundary detection. The following sections describe some typical problems and our solutions.

2.1 Problems with Dependency Structure Analysis

Ambiguous sentence boundaries

As described in Section 1, in this study, we assumed that ambiguous sentence boundaries is the biggest problem in dependency structure analysis of spontaneous speech. So in this paper, we mainly focus on this problem and describe our solution to it.

Independent *bunsetsu*s

In spontaneous speech, we sometimes find that modifiees are missing because utterance planning changes in the middle of the speech. Also, we sometimes find *bunsetsu*s whose dependency relationships are useless for understanding the utterance. These include fillers such as "あのー (*anoh*, well)" and "そのー (*sonoh*, well)", adverbs that behave like fillers such as "もう (*mou*)", responses such as "はい (*hai*, yes)" and " うん (*un*, yes)", conjunctions such as "で (*de*, and)", and disfluencies. In these cases, *bunsetsu*s are assumed to be independent, and as a result, they have no modifiees in the CSJ. For example, 14,988 *bunsetsu*s in 188 talks in the CSJ are independent.

We cannot ignore fillers, responses, and disfluencies because they frequently appear in spontaneous speech. However, we can easily detect them by using the method proposed by Asahara and Matsumoto (Asahara and Matsumoto, 2003). In this paper, fillers, responses, and disfluencies were eliminated before dependency structure analysis and sentence boundary detection by using morphological information and labels. In the CSJ, fillers and responses are interjections, and almost all of them are marked with label (F). Disfluencies are marked with label (D).

In this paper, every independent *bunsetsu* was assumed to depend on the next one. However, practically speaking, independent *bunsetsu*s should be correctly detected as "independent". This detection is one of our future goals.

Crossed dependency

In general, dependencies in Japanese written text do not cross. In contrast, dependencies in spontaneous speech sometimes do. For example, "これが (*kore-ga*, this)" depends on "正しいと (*tadashii-to*, is right)" and "私は (*watashi-wa*, I)" depends on "思う (*omou*, think)" in the sentence "これが／私は／正しいと／思う", where "／" denotes a *bunsetsu* boundary. Therefore, the two dependencies cross.

However, there are few number of crossed dependencies in the CSJ: In 188 talks, we found 689 such dependencies for total of 170,760 *bunsetsu*s. In our experiments, therefore, we assumed that dependencies did not cross. Correctly detecting crossed dependencies is one of our future goals.

Self-correction

We often find self-corrections in spontaneous speech. For example, in the 188 talks in the CSJ there were 2,544 self-corrections. In the CSJ, self-corrections are represented as dependency relationships between *bunsetsu*s, and label D is assigned to them. Coordination and appositives are also represented as dependency relationships between *bunsetsu*s, and labels P and A are assigned to them, respectively. The definitions of coordination and appositives follow those of the Kyoto University text corpus (Kurohashi and Nagao, 1997). Both the labels and the dependencies should be detected for applications such as automatic text summarization. However, in this study, we detected only the dependencies between *bunsetsu*s, and we did it in the same manner as in previous studies using written text.

Inversion
Inversion occurs more frequently in spontaneous speech than in written text. For example, in the 188 talks in the CSJ there were 172 inversions. In the CSJ, inversions are represented as dependency relationships going in the direction from right to left. In this study, we thought it important to detect dependencies, and we manually changed their direction to that from left to right. The direction of dependency has been changed to that from left to right.

2.2 Problems with Sentence Boundary Detection

In spontaneous Japanese speech, sentence boundaries are ambiguous. In the CSJ, therefore, sentence boundaries were defined based on clauses whose boundaries were automatically detected using surface information (Maruyama et al., 2003), and they were detected manually (Takanashi et al., 2003). Clause boundaries can be classified into the following three groups.

Absolute boundaries , or sentence boundaries in their usual meaning. Such boundaries are often indicated by verbs in their basic form.

Strong boundaries , or points that can be regarded as major breaks in utterances and that can be used for segmentation. Such boundaries are often indicated by clauses whose rightmost words are "が (*ga*, but)", or "し (*shi*, and)".

Weak boundaries , or points that can be used for segmentation because they strongly depend on other clauses. Such boundaries are often indicated by clauses whose rightmost words are "ので (*node*, because)", or "たら (*tara*, if)".

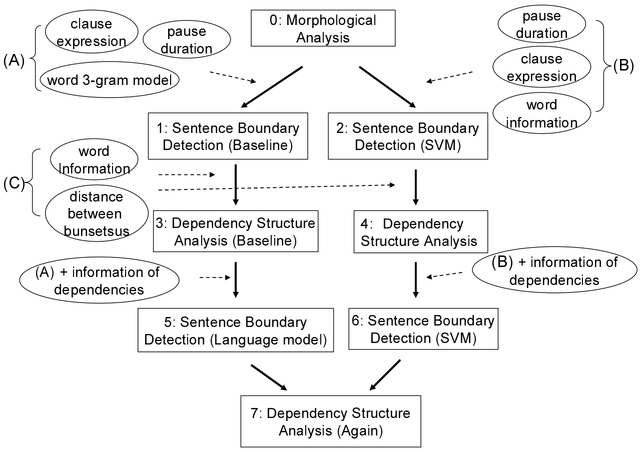
These three types of boundary differ in the degree of their syntactic and semantic completeness and the dependence of their subsequent clauses. Absolute boundaries and strong boundaries are usually defined as sentence boundaries. However, sentence boundaries in the CSJ are different from these two types of clause boundaries, and the accuracy of rule-based automatic sentence boundary detection in the 188 talks in the CSJ has an F-measure of approximately 81, which is the accuracy for a closed test. Therefore, we need a more accurate sentence boundary detection system.

Shitaoka et al. (Shitaoka et al., 2002) proposed a method for detecting sentence boundaries in spontaneous Japanese speech. Their definition of sentence boundaries is approximately the same as that of absolute boundaries described above. In this method, sentence boundary candidates are extracted by character-based pattern matching using pause duration. However, it is difficult to extract appropriate candidates by this method because there is a low correlation between pauses and the strong and weak boundaries described above. It is also hard to detect noun-final clauses by character-based pattern matching.

One method based on machine learning, a method based on maximum entropy models, has been proposed by Reynar and Ratnaparkhi (Reynar and Ratnaparkhi, 2000). However, the target in their study was written text. This method cannot readily used for spontaneous speech because in speech, there are no punctuation marks such as periods. Other features of utterances should be used to detect sentence boundaries in spontaneous speech.

3 Approach of Dependency Structure Analysis and Sentence Boundary Detection

The outline of the processes is shown in Figure 1.



(A)
clause expression
pause duration
word 3-gram model
0: Morphological Analysis
pause duration
clause expression
word information
(B)
1: Sentence Boundary Detection (Baseline)
2: Sentence Boundary Detection (SVM)
(C)
word information
distance between bunsetsus
3: Dependency Structure Analysis (Baseline)
4: Dependency Structure Analysis
(A) + information of dependencies
(B) + information of dependencies
5: Sentence Boundary Detection (Language model)
6: Sentence Boundary Detection (SVM)
7: Dependency Structure Analysis (Again)

Figure 1: Outline of dependency structure analysis and sentence boundary detection.

3.1 Dependency Structure Analysis

In statistical dependency structure analysis of Japanese speech, the likelihood of dependency is represented by a probability estimated by a dependency probability model.

Given sentence S, let us assume that it is uniquely divided into n *bunsetsu*s, $b_1, \ldots, b_n$, and that it is represented as an ordered set of *bunsetsu*s, $B = \{b_1, \ldots, b_n\}$. Let D be an ordered set of dependencies in the sentence and let D_i be a dependency whose modifier is *bunsetsu* $b_i (i = 1, \ldots, n-1)$. Let us also assume that $D = \{D_1, \ldots, D_{n-1}\}$. Statistical dependency structure analysis finds dependencies that maximize probability $P(D|S)$ given sentence S.

The conventional statistical model (Collins, 1996; Fujio and Matsumoto, 1998; Haruno et al., 1998; Uchimoto et al., 1999) uses only the relationship between two *bunsetsu*s to estimate the probability of dependency, whereas the model in this study (Uchimoto et al., 2000) takes into account not only the relationship between two *bunsetsu*s but also the relationship between the left *bunsetsu* and all the *bunsetsu*s to its right. This model uses more information than the conventional model.

We implemented this model within a maximum entropy modeling framework. The features used in the model were basically attributes of *bunsetsu*s, such as character strings, parts of speech, and types of inflections, as well as those that describe the relationships between *bunsetsu*s, such as the distance between *bunsetsu*s. Combinations of these features were also used. To find D_{best}, we analyzed the sentences backwards (from right to left). In the backward analysis, we can limit the search space effectively by using a beam search. Sentences can also be analyzed deterministically without great loss of accuracy (Uchimoto et al., 1999). So we analyzed a sentence backwards and deterministically.

3.2 Sentence Boundary Detection Based on Statistical Machine Translation (Conventional method (Shitaoka et al., 2002))

The framework for statistical machine translation is formulated as follows. Given input sequence X, the goal of statistical machine translation is to find the best output sequence, Y, that maximizes conditional probability $P(Y|X)$:

$$\max_Y P(Y|X) = \max_Y P(Y)P(X|Y) \qquad (1)$$

The problem of sentence boundary detection can be reduced to the problem of translating a sequence of words, X, that does not include periods but instead includes pauses into a sequence of words, Y, that includes periods. Specifically, in places where a pause might be converted into a period, which means $P(X|Y) = 1$, the decision whether a period should be inserted or not is made by comparing language model scores $P(Y')$ and $P(Y'')$. Here, the difference between Y' and Y'' is in that one includes a period in a particular place and the other one does not.

We used a model that uses pause duration and surface expressions around pauses as translation model $P(X|Y)$. We used expressions around absolute and strong boundaries as described in Section 2.2 as surface expressions around pauses. A pause preceding or following surface expressions can be converted into a period. Specifically, pauses following expressions "と (*to*)", "ない (*nai*)", and "た (*ta*)", and pauses preceding expression "で (*de*)", can be converted into a period when these pauses are longer than average. A pause preceding or following other surface expressions can be converted into a period even if its duration is short. To calculate $P(Y)$, we used a word 3-gram model trained with transcriptions in the CSJ.

3.3 Sentence Boundary Detection Using Dependency Information (Method 1)

There are three assumptions that should be satisfied by the rightmost *bunsetsu* in every sentence. In the following, this *bunsetsu* is referred to as the target *bunsetsu*.

(1) One or more *bunsetsu*s depend on the target *bunsetsu*. (Figure 2)

Since every *bunsetsu* depends on another *bunsetsu* in the same sentence, the second rightmost *bunsetsu* always depends on the rightmost *bunsetsu* in any sentence, except in inverted sentences. In inverted sentences in this study, we changed the direction of all dependencies to that from left to right.

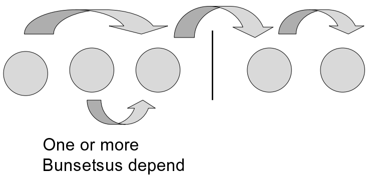


Figure 2: One or more *bunsetsu*s depend on the target *bunsetsu*. ("|" represents a sentence boundary.)

(2) There is no *bunsetsu* that depends on a *bunsetsu* beyond the target *bunsetsu*. (Figure 3)

Each *bunsetsu* in a sentence depends on a *bunsetsu* in the same sentence.

(3) The probability of the target *bunsetsu* is low. (Figure 4)

The target *bunsetsu* does not depend on any *bunsetsu*.

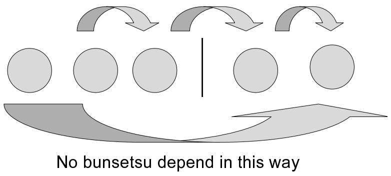


Figure 3: There is no *bunsetsu* that depends on a *bunsetsu* beyond the target *bunsetsu*.

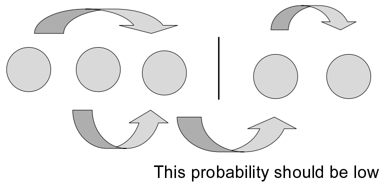


Figure 4: Probability of the target *bunsetsu* is low.

*Bunsetsu*s that satisfy assumptions (1)-(3) are extracted as rightmost *bunsetsu* candidates in a sentence. Then, for every point following the extracted *bunsetsu*s and for every pause preceding or following the expressions described in Section 3.2, a decision is made regarding whether a period should be inserted or not. In assumption (2), *bunsetsu*s that depend on a *bunsetsu* beyond 50 *bunsetsu*s are ignored because no such long-distance dependencies were found in the 188 talks in the CSJ used in our experiments. *Bunsetsu*s whose dependency probability is very low are also ignored because there is a high possibility that these *bunsetsu*s' dependencies are incorrect. Let this threshold probability be p, and let the threshold probability in assumption (3) be q. The optimal parameters p and q are determined by using held-out data.

In this approach, about one third of all *bunsetsu* boundaries are extracted as sentence boundary candidates. So, an output sequence is selected from all possible conversion patterns generated using two words to the left and two words to the right of each sentence boundary candidate. To perform this operation, we used a beam search with a width of 10 because a number of conversion patterns can be generated with such a search.

3.4 Sentence Boundary Detection Based on Machine Learning (Method 2)

We use Support Vector Machine (SVM) as a machine learning model and we approached the problem of sentence boundary detection as a text chunking task. We used YamCha (Kudo and Matsumoto, 2001) as a text chunker, which is based on SVM and uses polynomial kernel functions. To determine the appropriate chunk label for a target word, YamCha uses two words to the right and two words to the left of the target word as statistical features, and it uses chunk labels that are dynamically assigned to the two preceding or the two following words as dynamic features, depending on the analysis direction. To solve the multi-class problem, we used pairwise classification. This method generates $N * (N - 1)/2$ classifiers for all pairs of classes, N, and makes a final decision by their weighted voting.

The features used in our experiments are the following:

1. Morphological information of the three words to the right and three words to the left of the target word, such as character strings, pronunciation, part of speech, type of inflection, and inflection form
2. Pause duration normalized in terms of Mahalanobis distance
3. Clause boundaries
4. Dependency probability of the target *bunsetsu*
5. The number of *bunsetsu*s that depend on the target *bunsetsu* and their dependency probabilities

We used the IOE labeling scheme for proper chunking, and the following parameters for YamCha.

- Degree of polynomial kernel: 3rd
- Analysis direction: Left to right
- Multi-class method: Pairwise

4 Experiments and Discussion

In our experiments, we used the transcriptions of 188 talks in the CSJ. We used 10 talks for testing. Dependency structure analysis results were evaluated for closed- and open-test data in terms of accuracy, which was defined as the percentage of correct dependencies out of all dependencies. In Tables 1 to 3, we use words "closed" and "open" to describe the results obtained for closed- and open-test data, respectively. Sentence boundary detection results were evaluated in terms of F-measure.

First, we show the baseline accuracy of dependency structure analysis and sentence boundary detection. The method described in Section 3.2 was used as a baseline method for sentence boundary detection (Process 1 in Figure 1). To train the language model represented by $P(Y)$, we used the transcriptions of 178 talks excluding the test data. The method described in Section 3.1 was used as a baseline method for dependency structure analysis. (Process 3 in Figure 1) As sentence boundaries, we used the results of the baseline method for sentence boundary detection. We obtained an F-measure of 75.6, a recall of 64.5%, and a precision of 94.2% for the sentence boundary detection in our experiments. The dependency structure analysis accuracy was 75.2% for the open data and 80.7% for the closed data.

The dependency probability of the rightmost *bunsetsu*s in a given sentence was not calculated in our model. So, we assumed that the rightmost *bunsetsu*s depended on the next *bunsetsu* and that the dependency probability was 0.5 when we used dependency information in the experiments described in the following sections.

4.1 Sentence Boundary Detection Results Obtained by Method 1

We evaluated the results obtained by the method described in Section 3.3. The results of baseline dependency structure analysis were used as dependency information (Process 5 in Figure 1).

First, we investigated the optimal values of parameters p and q described in Section 3.3 by using held-out data, which differed from the test data and consisted of 15 talks. The optimal values of p and q were, respectively, 0 and 0.9 for the open-test data, and 0 and 0.8 for the closed-test data. These values were used in the following experiments. The value of p was 0, and these results show that *bunsetsu*s that depended on a *bunsetsu* beyond 50 *bunsetsu*s were ignored as described in assumption (2) in Section 3.3.

The obtained results are shown in Table 1. When dependency information was used, the F-measure increased by approximately 1.4 for the open-test data and by 2.0 for the closed test data, respectively. Although the accuracy of dependency structure analysis for closed test data was about 5.5% higher than that for the open-test data, the difference between the accuracies of sentence boundary detection for the closed- and open-test data was only about 0.6%. These results indicate that equivalent accuracies can be obtained for both open- and closed-test data in detecting dependencies related to sentence boundaries.

When all the extracted candidates were considered as sentence boundaries without using language models, the accuracy of sentence boundary detection obtained by using the baseline method was 68.2%(769/1,127) in recall and 81.5%(769/943) in precision, and that obtained by using Method 1 was 87.2%(983/1,127) in recall and 27.7%(983/3,544) in precision. The results show that additional 214 sentence boundary candidates were correctly extracted by using dependency information. However, only 108 sentence boundaries were chosen out of the 214 candidates when language models were used. We investigated in detail the points that were not chosen and found errors in noun-final clauses, clauses where the rightmost constituents were adjectives or verbs such as "と思う (it to-omou, think)" or "は難しい (it wa-muzukashii, difficult)", and clauses where the rightmost constituents were "というのは (it to-iu-no-wa, because)" and "としては (it to-si-te-wa, as)", and so on. Some errors, except for those in noun-final clauses, could have been correctly detected if we had had more training data.

Table 1: Sentence boundary detection results obtained by using dependency information.

	recall	precision	F
With dependency information (open)	74.1% (835/1,127)	82.5% (835/1,012)	78.0
With dependency information (closed)	74.2% (836/1,127)	83.5% (836/1,001)	78.6
baseline	64.5% (727/1,127)	94.2% (727/772)	76.6

We also found that periods were sometimes erroneously inserted when preceding expressions were "が (*ga*, but)", "まして (*mashite*, and)", and "けれども (*keredomo*, but)", which are typically the rightmost constituents of a sentence, as weel as "て (*te*, and)", which is not, typically, the rightmost constituent of a sentence. The language models were not good at discriminating between subtle differences.

4.2 Sentence Boundary Detection Results Obtained by Method 2

We evaluated the results obtained by the method described in Section 3.4 (Process 6 in Figure 1). For training, we used 178 talks excluding test data.

The results are shown in Table 2. The F-measure was about 6.9 points higher than that described in Section 4.1. The results show that the approach based on machine learning is more effective than that based on statistical machine translation. The results also show that the accuracy of sentence boundary detection can be increased by using dependency information in Method 2. However, we found that the amount of accuracy improvement achieved by using dependency information depended on the method used. This may be because other features used in SVM may provide information similar to dependency information. For example, Feature 1 described in Section 3.4 might provide information similar to that in Features 4 and 5. Although in our experiments we used only three words to the right and three words to the left of the target word, the degradation in accuracy without dependency information was slight. This may be because long-distance dependencies may not be related to sentence boundaries, or because Feature 5 does not contribute to increasing the accuracy because the accuracy of dependency structure analysis in detecting long-distance dependencies is not high.

Table 2: Sentence boundary detection results obtained by using SVM.

	recall	precision	F
With dependency information (open)	80.0% (902/1,127)	90.3% (902/999)	84.9
With dependency information (closed)	79.7% (900/1,127)	90.5% (900/994)	84.9
Without dependency information	79.3% (894/1,127)	90.1% (894/992)	84.4

Table 3: Dependency structure analysis results obtained with automatically detected sentence boundaries.

	open	closed
With results in Section 4.1	75.8%	81.2%
With results in Section 4.2	77.2%	82.5%
Baseline	75.2%	80.7%

4.3 Dependency Structure Analysis Results

We evaluated the results of dependency structure analysis obtained when sentence boundaries detected automatically by the two methods described above were used as inputs (Process 7 in Figure 1). The results are shown in Table 3. The accuracy of dependency structure analysis improved by about 2% when the most accurate and automatically detected sentence boundaries were used as inputs. This is because more sentence boundaries were detected correctly, and the number of *bunsetsu*s that depended on those in other sentences decreased.

We investigated the accuracy of dependency structure analysis when 100% accurate sentence boundaries were used as inputs. The accuracy was 80.1% for the open-test data, and 86.1% for the closed-test data. Even when the sentence boundary detection was perfect, the error rate was approximately 14% even for the closed-test data. The accuracy of dependency structure analysis for spoken text was about 8% lower than that for written text (newspapers). We speculate that this is because spoken text has no punctuation marks and many *bunsetsu*s depend on others far from them because of insertion structures. These problems need to be addressed in future studies.

5 Conclusion

This paper described a project to detect dependencies between *bunsetsu*s and sentence boundaries in a spontaneous speech corpus. It is more difficult to detect dependency structures in spontaneous spoken speech than in written text. The biggest problem is that sentence boundaries are ambiguous. We proposed two methods for improving the accuracy of sentence boundary detection in spontaneous Japanese speech. Using these methods, we obtained an F-measure of 84.9 for the accuracy of sentence boundary detection. The accuracy of dependency structure analysis was also improved from 75.2% to 77.2% by using automatically detected sentence boundaries. The accuracy of dependency structure analysis and that of sentence boundary detection were improved by interactively using automatically detected dependency information and sentence boundaries.

There are several future directions. In the future, we would like to solve the problems that we found in our experiments. In particular, we want to reduce the number of errors due to inserted structures and solve other problems described in Section 2.1.

References

Masayuki Asahara and Yuji Matsumoto. 2003. Filler and Disfluency Identification Based on Morphological Analysis and Chunking. In *Proceedings of the ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, pages 163–166.

Michael Collins. 1996. A New Statistical Parser Based on Bigram Lexical Dependencies. In *Proceedings of the ACL*, pages 184–191.

Masakazu Fujio and Yuji Matsumoto. 1998. Japanese Dependency Structure Analysis based on Lexicalized Statistics. In *Proceedings of the EMNLP*, pages 87–96.

Masahiko Haruno, Satoshi Shirai, and Yoshifumi Ooyama. 1998. Using Decision Trees to Construct a Practical Parser. In *Proceedings of the COLING-ACL*, pages 505–511.

Taku Kudo and Yuji Matsumoto. 2000. Japanese Dependency Structure Analysis Based on Support Vector Machines. In *Proceedings of the EMLNP*, pages 18–25.

Taku Kudo and Yuji Matsumoto. 2001. Chunking with support vector machines. In *Proceedings of the NAACL*.

Sadao Kurohashi and Makoto Nagao. 1997. Building a Japanese Parsed Corpus while Improving the Parsing System. In *Proceedings of the NLPRS*, pages 451–456.

Kikuo Maekawa, Hanae Koiso, Sadaoki Furui, and Hitoshi Isahara. 2000. Spontaneous Speech Corpus of Japanese. In *Proceedings of the LREC2000*, pages 947–952.

Takehiko Maruyama, Hideki Kashioka, Tadashi Kumano, and Hideki tanaka. 2003. Rules for Automatic Clause Boundary Detection and Their Evaluation. In *Proceedings of the Nineth Annual Meeting of the Association for Natural Language proceeding*, pages 517–520. (in Japanese).

Shigeki Matsubara, Takahisa Murase, Nobuo Kawaguchi, and Yasuyoshi Inagaki. 2002. Stochastic Dependency Parsing of Spontaneous Japanese Spoken Language. In *Proceedings of the COLING2002*, pages 640–645.

Jeffrey C. Reynar and Adwait Ratnaparkhi. 2000. A Maximum Entropy Approach to Identifying Sentence Boundaries. In *Proceedings of the ANLP*, pages 16–19.

Kazuya Shitaoka, Tatsuya Kawahara, and Hiroshi G. Okuno. 2002. Automatic Transformation of Lecture Transcription into Document Style using Statistical Framework. In *IPSJ–WGSLP SLP-41-3*, pages 17–24. (in Japanese).

Katsuya Takanashi, Takehiko Maruyama, Kiyotaka Uchimoto, and Hitoshi Isahara. 2003. Identification of "Sentences" in Spontaneous Japanese — Detection and Modification of Clause Boundaries —. In *Proceedings of the ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, pages 183–186.

Kiyotaka Uchimoto, Satoshi Sekine, and Hitoshi Isahara. 1999. Japanese Dependency Structure Analysis Based on Maximum Entropy Models. In *Proceedings of the EACL*, pages 196–203.

Kiyotaka Uchimoto, Masaki Murata, Satoshi Sekine, and Hitoshi Isahara. 2000. Dependency Model Using Posterior Context. In *Proceedings of the IWPT*, pages 321–322.