Construction of linefeed insertion rules for lecture transcript and their evaluation

Masaki Murata*

Department of Systems and Social Informatics,
Graduate School of Information Science,
Nagoya University,
Furo-cho, Chikusa-ku, 464-8601, Japan
E-mail: murata@el.itc.nagoya-u.ac.jp
*Corresponding author

Tomohiro Ohno

Graduate School of International Development,
Nagoya University,
Furo-cho, Chikusa-ku, 464-8601, Japan
E-mail: ohno@nagoya-u.jp

Shigeki Matsubara

Information Technology Center,
Nagoya University,
Furo-cho, Chikusa-ku, 464-8601, Japan
E-mail: matubara@nagoya-u.jp

Abstract: The development of a captioning system that supports the real-time understanding of monologue speech such as lectures and commentaries is required. In monologues, since a sentence tends to be long, each sentence is often displayed in multi lines on the screen. In the case, it is necessary to insert linefeeds into a text so that the text becomes easy to read. This paper proposes a rule-based technique for inserting linefeeds into a Japanese spoken monologue sentence as an elemental technique to generate the readable captions. Our method inserts linefeeds into a sentence by applying the rules based on morphemes, dependencies and clause boundaries. We established the rules by circumstantially investigating the corpus annotated with linefeeds. An experiment using Japanese monologue corpus has shown the effectiveness of our rules.

Keywords: spoken language; sentence analysis; real-time captioning; clause boundary; speech corpus.

Reference to this paper should be made as follows: Murata, M., Ohno, T. and Matsubara, S. (xxxx) 'Construction of linefeed insertion rules for lecture transcript and their evaluation', *Int. J. Knowledge and Web Intelligence*, Vol. x, No. x, pp.xxx–xxx.

Biographical notes: Masaki Murata received his BE degree in Information Engineering from the Nagoya University, Japan, in 2008. Currently, he is a master course student at the Nagoya University. His research interests include natural language processing.

Tomohiro Ohno received the Dr. of Information Science from the Nagoya University in 2007. He was a Research Fellow of the JSPS from 2006 to 2007. Since 2007, he has been an Assistant Professor of the Graduate School of International Development, Nagoya University. His research interests include natural language processing and spoken language processing. He is a Member of the ACL.

Shigeki Matsubara received the Dr. of Engineering from the Nagoya University in 1998. He was a Research Fellow of the JSPS from 1996 to 1998, and a Research Associate from 1998 to 2002 at the Faculty of Language and Culture, Nagoya University. Since 2002, he has been an Associate Professor of the Information Technology Center, Nagoya University. His research interests include natural language processing, spoken language processing and digital library. He is a Member of the IEEE, the ACM and the ACL.

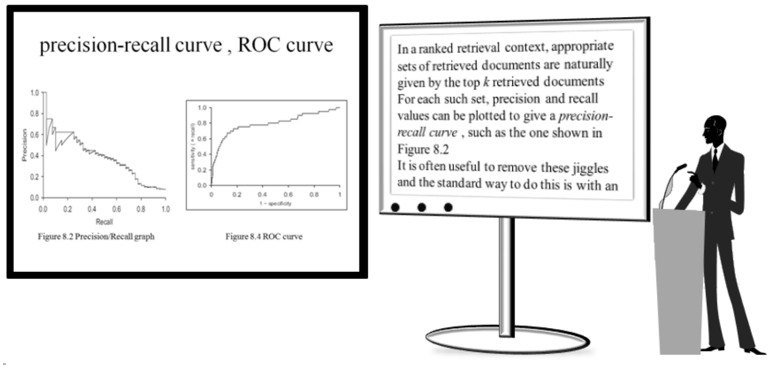
1 Introduction

Real-time captioning, which displays transcribed texts of monologue speech such as lectures, is a technique for supporting the speech understanding of deaf persons, elderly persons, or foreigners. In monologues, since a sentence tends to be long, each sentence is often displayed in multi lines on the screen. In the case, it is necessary to insert linefeeds into a text so that the text becomes easy to read.

This paper proposes a technique for inserting linefeeds into a Japanese spoken monologue sentence as an elemental technique to generate readable captions. We assume that a screen which displays only multiline captions to provide the caption information to the audience is placed on the site of lectures and commentaries. In our method, the linefeeds are assumed to be inserted into only the boundaries between *bunsetsus*[1]. Our method applies the rules for inserting linefeeds to a sentence. The rules are created in consideration of the boundary into which a linefeed is not inserted, the boundary into which a linefeed should be inevitably inserted, and the boundary into which a linefeed can be inserted.

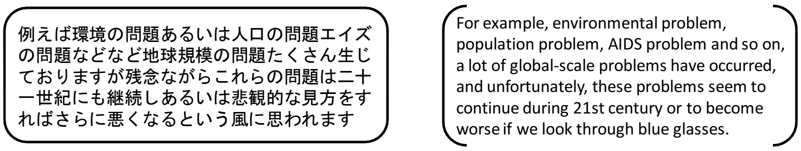
We established the rules based on the emerging pattern of morphemes, dependencies and clause boundaries by circumstantially investigating the corpus annotated with linefeeds. We conducted an experiment on inserting linefeeds by using a Japanese spoken monologue corpus. As the results, the precision and recall of our method was 80.2% and 67.4%, respectively. Our method improved the performance dramatically compared with the baseline method, which is implemented based on bunsetsu boundaries and the maximum number of characters per line, and has been confirmed to be effective.

Figure 1 Caption display of spoken monologue


precision-recall curve , ROC curve
Precision
Recall
Figure 8.2 Precision/Recall graph
Figure 8.4 ROC curve

In a ranked retrieval context, appropriate sets of retrieved documents are naturally given by the top *k* retrieved documents For each such set, precision and recall values can be plotted to give a *precision-recall curve* , such as the one shown in Figure 8.2
It is often useful to remove these jiggles and the standard way to do this is with an

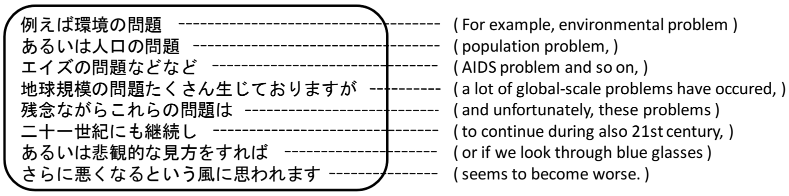
Figure 2 Caption of monologue speech



例えば環境の問題あるいは人口の問題エイズの問題などなど地球規模の問題たくさん生じておりますが残念ながらこれらの問題は二十一世紀にも継続しあるいは悲観的な見方をすればさらに悪くなるという風に思われます

For example, environmental problem, population problem, AIDS problem and so on, a lot of global-scale problems have occurred, and unfortunately, these problems seem to continue during 21st century or to become worse if we look through blue glasses.

Figure 3 Caption into which linefeeds are properly inserted



例えば環境の問題 (For example, environmental problem)
あるいは人口の問題 (population problem,)
エイズの問題などなど (AIDS problem and so on,)
地球規模の問題たくさん生じておりますが (a lot of global-scale problems have occured,)
残念ながらこれらの問題は (and unfortunately, these problems)
二十一世紀にも継続し (to continue during also 21st century,)
あるいは悲観的な見方をすれば (or if we look through blue glasses)
さらに悪くなるという風に思われます (seems to become worse.)

This paper is organized as follows: The next section describes our assumed caption. Section 3 shows the analysis to make the rules. Section 4 presents our linefeed insertion technique. The experiment and discussion are reported in Section 5.

2 Caption display of spoken monologue

2.1 Linefeeds insertion in monologue sentences

In our research, as an environment in which captions are displayed on the site of lectures, we assume that a screen for displaying only captions is used. Figure 1 shows our assumed environment in which captions are displayed. In the screen, multi lines are always displayed, being scrolled line by line.

As shown in Figure 2, if the transcribed text of monologue speech is simply displayed in accordance with only the width of the screen without considering the proper points of linefeeds, the caption becomes hard to read. Especially, since the audience are forced to read the caption in accordance with the speaker's utterance speed, it is important that linefeeds are properly inserted into the displayed text in consideration of the good readability as shown in Figure 3.

In our research, we set the following concepts as the proper points into which linefeeds are inserted on captioning.

- Linefeeds have to be inserted so that each line constitutes a semantically meaningful unit.
- The number of characters in each line has to be less than or equal to the maximum number of characters per line, which is established based on the width of a screen.

Here, since a bunsetsu is the smallest semantically meaningful language unit in Japanese, our method adopts the bunsetsu boundaries as the candidates of points into which linefeeds are inserted. In this paper, hereafter, we call a bunsetsu boundary into which a linefeed should be inserted a **linefeed point**.

2.2 Related works

There exist a lot of researches about captioning, and the techniques of automatic speech recognition (ASR) aimed for captioning have been developed (Boulianne et al., 2006; Daelemans, Hothker and Sang, 2004; Holter et al., 2000; Imai et al., 2006; Munteanu, Penn and Baecker, 2007; Saraclar et al., 2002; Xue, Hu and Zhao, 2006). However, in order to generate captions which are easy to read, it is important not only to recognize speech with high recognition rate but also to properly display the transcribed text on a screen (Nakano et al., 2007). There are few conventional researches about inserting linefeeds on captioning except the following researches. Monma et al. (2003) proposed the method for inserting linefeeds based on patterns of a sequence of morphemes. They analyzed the point into which linefeeds were inserted on the closed-captions of Japanese TV shows, and then made the rules for inserting linefeeds. However, in this research, the linefeeds are inserted on the constraint that the text displayed in a screen all switches to the next text at a time, that is, the readability in case of our assumed caption display system is not considered.

Saiko, Takanashi and Kawahara (2005) proposed the method for captioning based on the gradual chunking. This method chunks morphemes into "*constituents*," which corresponds to the nominative, predicates, case elements and so on in a sentence, and then chunks "*constituents*" into "*phrases*." In the application of the chunking model to linefeed insertion, the concatenation of *constituents* is reiterated until the length of each line reaches 15 characters, and then, a linefeed is inserted just before the length of the line exceeds 15 characters. In this regard, however, a linefeed is unconditionally inserted between neighboring *phrases*. Although this method inserts linefeeds so that each line becomes a linguistic unit, the research did not verify the relation between linefeed points and the *constituent* or *phrase*.

Ohno, Murata and Matsubara (2009) proposed a linefeed insertion technique based on machine learning. The technique uses a monologue corpus annotated

Table 1 Size of the analysis data

sentences	221
bunsetsus	2,891
characters	13,899
linefeeds	833
characters per line	13.2

with the information on proper linefeed insertion, and decides the linefeed points by adopting pauses, clause boundaries, dependencies, etc. as features for machine learning. Using a large-scale corpus enables the technique to insert the linefeeds with high accuracy. However, the performance of linefeeds insertion depends heavily on the size and the feature of the learning data. On the other hand, we are aiming at construction of general-purpose rules by manually investigating lecture transcripts. These rules suggest linguistic and acoustic factors which are related to the linefeed points, and are significant from the viewpoint of linguistics.

3 Linefeeds in monologue sentences

We investigated the actual spoken monologue data to make the rules for inserting linefeeds based on the concepts described in Section 2.1. In our investigation, we used Japanese monologue speech data in the simultaneous interpretation database (Matsubara et al., 2002). The morphological analysis, bunsetsu segmentation, clause boundary analysis and dependency analysis are performed automatically[2] on this data, and then those information are modified by hand[3]. Table 1 shows the size of the analysis data. In what follows, we organize bunsetsu boundaries by classifying them into the following three categories: the boundary into which a linefeed is not inserted, the boundary into which a linefeed should be inevitably inserted, and the boundary into which a linefeed can be inserted.

3.1 Boundaries into which linefeeds are not inserted

As the result of the investigation, we observed that linefeeds were hardly inserted into the following bunsetsu boundaries.

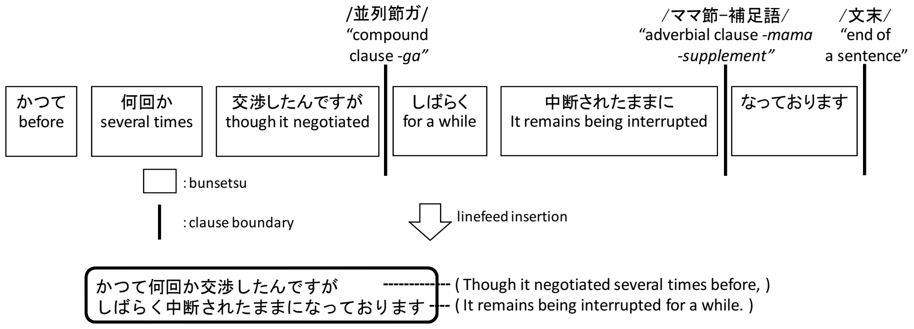
- The end boundary of the bunsetsu of which the part-of-speech of the rightmost morpheme is “adnominal,” “adverb-particle_conjunction,” “particle-adnominalizer,” “noun-adverbial,” “adjective-main,” or, which the basic form of the rightmost morpheme is “あるいは (or)”, “と (ϕ)”.
- The start boundary of the bunsetsu of which the part-of-speech of the leftmost morpheme is “noun-affix-misc,” “noun-*nai*_adjective,” “noun-affix-adverbial,” or, which the basic form of the leftmost morpheme is “思う (think)”, “問題 (problem)”, “する (do)”, “なる (become)”, “必要 (necessary)”.

For example, the bunsetsu “実際の (actual)” has the rightmost morpheme “の” of which part-of-speech is “adnominal particle.” Therefore, a linefeed is not

Table 2 Strong clause boundary

compound clause	*-ga*, *-shi*, *-keredomo*, *-de*
condition clause	*-baai*, *-tokoro*
time clause	*-tokini*
reason clause	*-kara*
adverbial clause	*-tameniha*, *-tameni*, *-yo*, *-temo*, *-tekara*
others	continuous clause, indirect interrogative, indirect interrogative -supplement, supplement clause -conjunction, subordinate sentence, quotational clause-particle, adnominal adjective clause

Figure 4 Relation between strong clause boundary and a linefeed point

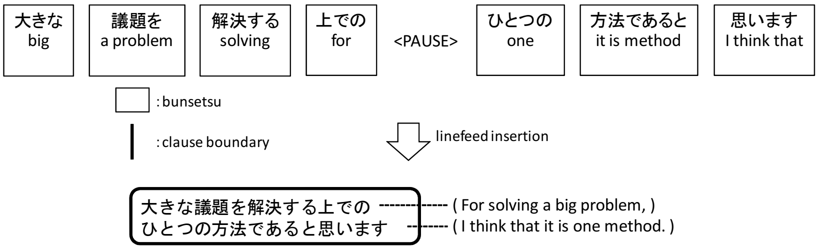


inserted between the sequence of two bunsetsus "実際の (actual)" and "説明会に (explanatory meeting)."

3.2 Boundaries into which linefeeds are inevitably inserted

In Japanese, a clause basically contains one verb phrase and consists of a sequence of bunsetsus. Since a clause constitutes a semantically meaningful language unit, bunsetsu boundaries which are clause boundaries can be widely-accepted as the candidate of a linefeed point. However, the role of each clause of a sentence is different by the type. This means that the likelihood that a linefeed is inserted into a clause boundary is different by the type of the clause boundary. As the result of the above-mentioned analysis, there existed 20 types[4] of clause boundaries into which linefeeds should be inevitably inserted. Table 2 shows the clause boundary types into which linefeeds should be inevitably inserted. We call these types of clause boundaries the **strong clause boundary** as a whole hereafter. Figure 4 shows the relation between strong clause boundary and linefeed points. The strong clause boundary accounted for 47.6% of clause boundaries which appear in the analysis data.

Figure 5 Relation between a pause and a linefeed point



3.3 Boundaries into which linefeeds can be inserted

3.3.1 Linefeeds insertion based on clause boundaries

There exist clause boundaries into which linefeeds are not necessarily inserted but are inserted with high probability in a context. As such a clause boundary type, there are "condition clause *-to*," "condition clause *-ba*," "reason clause *-node*," "compound clause *-te*," "compound clause *-toka*," "concessive clause *-temo*," "indeclinable words stopping," and "interjection." In this paper, we call these 8 types of a clause boundaries the **weak clause boundary**. The weak clause boundary becomes the point into which linefeed is inserted, if there does not exist the strong clause boundary around it.

3.3.2 Linefeeds insertion based on pauses

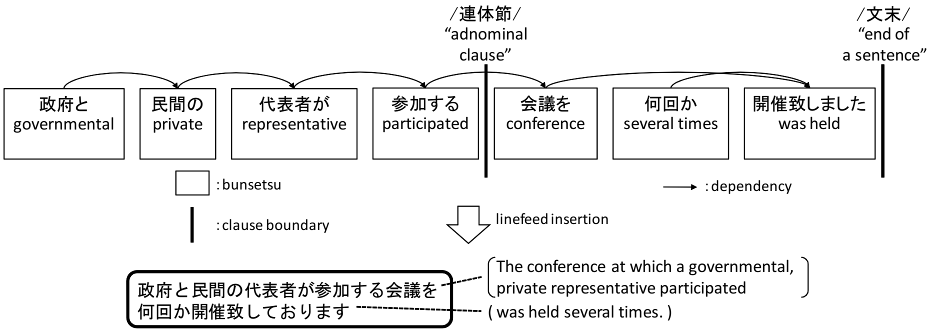
It is thought that a pause corresponds to a syntactic boundary (Iwata, Mitome and Watanabe, 1990). Therefore, there are possibility that a linefeed becomes more easily inserted into a bunsetsu boundary in which a pause exists. In our research, a pause is defined as a silent interval equal to or longer than 200ms. Figure 5 shows the relation between a pause and a linefeed point. In the analysis data, among 748 bunsetsu boundaries in which a pause exists, linefeeds were inserted into 471 bunsetsu boundaries, that is, the ratio of linefeed insertion was 63.0%. This ratio is higher than that of bunsetsu boundaries, thus, we confirmed that linefeeds tend to be inserted into bunsetsu boundaries in which a pause exists.

3.3.3 Linefeeds insertion based on dependency relations

A dependency relation is a modification relation in which a modifier bunsetsu depends on a modified bunsetsu. A sequence of bunsetsus from the modifying bunsetsu to the modified bunsetsu constitutes a semantically meaningful unit. Therefore, linefeeds tend to be inserted into the end boundaries of modified bunsetsus although the tendency is not greater than that of clause boundaries.

Among the end boundaries of modified bunsetsus, the end boundaries of modified bunsetsus of adnominal clauses have the strongest tendency for a linefeed to be inserted into. Figure 6 shows an example of linefeed insertion into the end boundaries of a modified bunsetsu of an adnominal clause. Here, in Japanese, the rightmost morpheme of an adnominal clause is congruent with that of a sentence

Figure 6 Linefeed insertion into the end boundaries of a modified bunsetsu of an adnominal clause



end. If a linefeed is inserted into the end boundary of adnominal clause, the end of the line may be misunderstood as a sentence end. Therefore a linefeed is inserted not there but into the end boundary of the modified bunsetsu of an adnominal clause.

Since a clause boundary labeled "topicalized element -*wa*" does not strictly represent a clause boundary but can be regarded as a syntactically independent element, it is the dominant candidate of the linefeed points (See Figure 7). However, when the number of characters from the start of a sentence to the clause boundary labeled "topicalized element -*wa*" is few, it is not appropriate to insert a linefeed into the boundary. If the length of the character string between the start of a line and the clause boundary labeled "topicalized element -*wa*" is long to some extent, a linefeed tends to be inserted into the clause boundary labeled "topicalized element -*wa*."

In addition, the dependency structure of a line displayed as a caption tends to be closed. That is to say, all the bunsetsus, except the final bunsetsu, in a line tend to depend on one of bunsetsus in the line. Conversely, a linefeed tends to be inserted into the end boundary of the modified bunsetsu of which the dependency distance is long (See Figure 8).

4 Linefeeds insertion rules

In our method, a sentence, on which the morphological analysis, bunsetsu segmentation, clause boundary analysis and dependency analysis are performed, is considered as the input. Our method outputs the sentence into which linefeeds are inserted. The insertion of linefeeds is executed as follows by using the rules for deciding the linefeed points.

We made the rules for inserting linefeeds based on the analysis described in the previous section. Table 3 shows the rules. The number for each rule indicates the priority order in which each rule is applied, and the application of rules is performed in accordance with the priority order until the length of all lines becomes less than or equal to the maximum number of characters per line.

Figure 7 Linefeed insertion into the clause boundary labeled “topicalized element *-wa*”

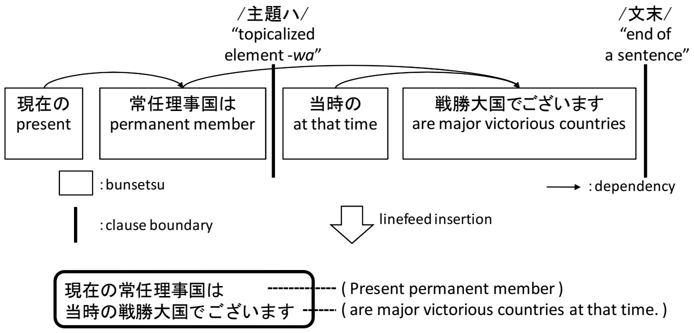
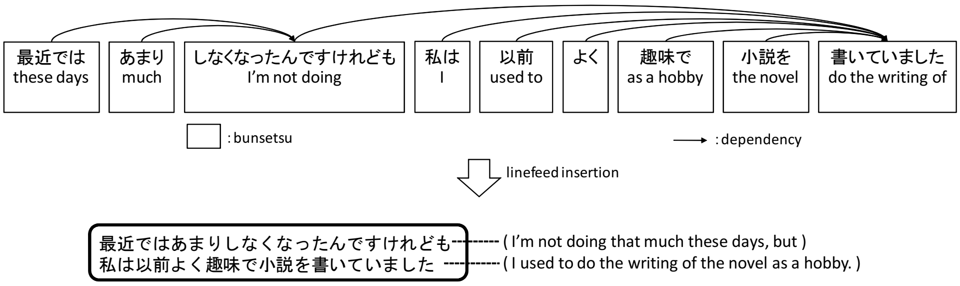


Figure 8 Linefeed insertion into the end boundary of a modifier bunsetsu


最近では
these days
あまり
much
しなくなったんですけれども
I’m not doing
私は
I
以前
used to
よく
趣味で
as a hobby
小説を
the novel
書いていました
do the writing of
: bunsetsu
: dependency
linefeed insertion
最近ではあまりしなくなったんですけれども
私は以前よく趣味で小説を書いていました
(I’m not doing that much these days, but)
(I used to do the writing of the novel as a hobby.)

The first rule detects the boundaries into which linefeeds are not inserted, and the second rule detects the boundaries into which linefeeds should be inevitably inserted. Furthermore, the rule 3 is the insertion based on clause boundaries, 4-5 and 7-8 are on dependency relations, 6 is on pauses, and 9 is on the number of characters of line.

Figure 9 shows the processing flow of linefeed insertion. The candidates of points into which linefeeds are inserted are denoted by a slash “/.” First, the end boundaries of the bunsetsus into which linefeeds are not inserted are excluded from the candidates of linefeed points. Next, linefeeds are inserted into the end boundary of the bunsetsu “生じておりますが (have occurred),” and “継続し (seem to continue),” which are respectively the strong clause boundary labeled “compound clause *-ga*” and “compound clause *-shi*.” As mentioned above, the rules for linefeed insertion are applied in accordance with the priority order. The texts of the caption are finally generated so that the length of each line is less than or equal to the maximum number of characters per line.

Table 3 Rules for inserting linefeeds

1	Exclude bunsetsus boundaries into which linefeeds are not inserted from the candidates.
2	Insert into the strong clause boundary.
3	Insert into the weak clause boundary.
4	Insert into the end boundary of a modifier bunsetsu of "adnominal clause."
5	Insert into the clause boundary labeled "topicalized element-*wa*," if the number of characters between the start of a line and "*wa*" is over 30% of the maximum number of characters per line.
6	Insert into the bunsetsu boundary in which a pause exists.
7	Insert into the end boundary of the **long dependency distance bunsetsu*** in case that there exits only one **long dependency distance bunsetsu** within the maximum number of characters from the start of the line.
8	Insert into the end boundary of the leftmost bunsetsu among **long dependency distance bunsetsus** of which the modified bunsetsu is different from that of the next one.
9	Insert into the rightmost bunsetsu boundary within the maximum number of characters.

*A bunsetsu which is located within the maximum number of characters from the start of the line and which depends on a bunsetsu located outside the maximum number of characters from the start of the line is called **long dependency distance bunsetsu**.

5 Experiment

To evaluate the effectiveness of our method, we conducted an experiment on inserting linefeeds by using Japanese spoken monologue data.

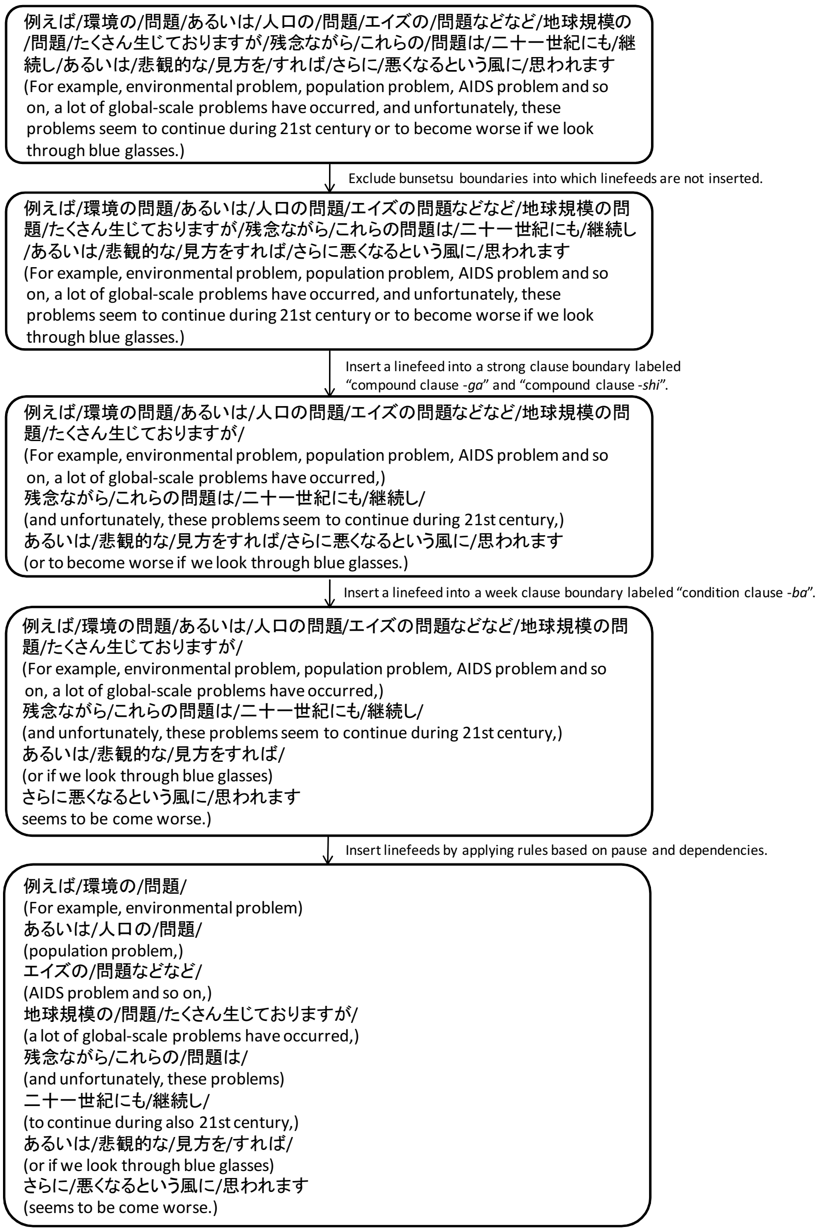
5.1 Outline of experiment

As the test data, we used the transcribed data of Japanese lecture speech (1,714 sentences, 18,993 bunsetsus, which were disjointed with 221 sentences in Section 3) in the simultaneous interpretation database (Matsubara et al., 2002). The all data are annotated by hand with information on the morphological analysis, clause boundary detection and dependency analysis.

We applied our method to the test data. In addition, we compared our method with the baseline one, which inserts linefeeds into the rightmost bunsetsu boundary among the bunsetsu boundaries into which linefeeds can be inserted so that the length of the line does not exceed the maximum number of characters. In the evaluation, we obtained the recall, the precision and the F-measure. The recall, precision and F-measure are respectively defined as follows.

$$recall = \frac{\#\ \ of\ \ correctly\ \ inserted\ \ linefeeds}{\#\ \ of\ \ linefeeds\ \ in\ \ the\ \ correct\ \ data}$$

Figure 9 Processing flow of linefeed insertion



例えば/環境の/問題/あるいは/人口の/問題/エイズの/問題などなど/地球規模の/問題/たくさん生じておりますが/残念ながら/これらの/問題は/二十一世紀にも/継続し/あるいは/悲観的な/見方を/すれば/さらに/悪くなるという風に/思われます
(For example, environmental problem, population problem, AIDS problem and so on, a lot of global-scale problems have occurred, and unfortunately, these problems seem to continue during 21st century or to become worse if we look through blue glasses.)

Exclude bunsetsu boundaries into which linefeeds are not inserted.

例えば/環境の問題/あるいは/人口の問題/エイズの問題などなど/地球規模の問題/たくさん生じておりますが/残念ながら/これらの問題は/二十一世紀にも/継続し/あるいは/悲観的な/見方をすれば/さらに悪くなるという風に/思われます
(For example, environmental problem, population problem, AIDS problem and so on, a lot of global-scale problems have occurred, and unfortunately, these problems seem to continue during 21st century or to become worse if we look through blue glasses.)

Insert a linefeed into a strong clause boundary labeled "compound clause *-ga*" and "compound clause *-shi*".

例えば/環境の問題/あるいは/人口の問題/エイズの問題などなど/地球規模の問題/たくさん生じておりますが/
(For example, environmental problem, population problem, AIDS problem and so on, a lot of global-scale problems have occurred,)
残念ながら/これらの問題は/二十一世紀にも/継続し/
(and unfortunately, these problems seem to continue during 21st century,)
あるいは/悲観的な/見方をすれば/さらに悪くなるという風に/思われます
(or to become worse if we look through blue glasses.)

Insert a linefeed into a week clause boundary labeled "condition clause *-ba*".

例えば/環境の問題/あるいは/人口の問題/エイズの問題などなど/地球規模の問題/たくさん生じておりますが/
(For example, environmental problem, population problem, AIDS problem and so on, a lot of global-scale problems have occurred,)
残念ながら/これらの問題は/二十一世紀にも/継続し/
(and unfortunately, these problems seem to continue during 21st century,)
あるいは/悲観的な/見方をすれば/
(or if we look through blue glasses)
さらに悪くなるという風に/思われます
seems to be come worse.)

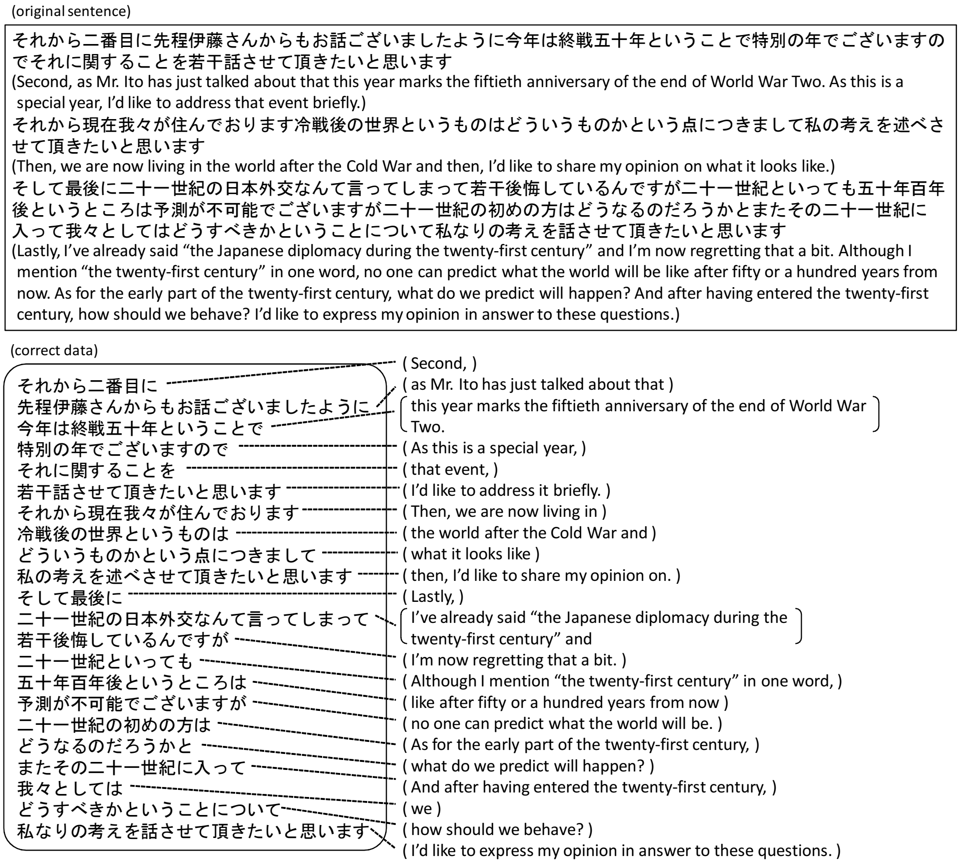
Insert linefeeds by applying rules based on pause and dependencies.

例えば/環境の/問題/
(For example, environmental problem)
あるいは/人口の/問題/
(population problem,)
エイズの/問題などなど/
(AIDS problem and so on,)
地球規模の/問題/たくさん生じておりますが/
(a lot of global-scale problems have occurred,)
残念ながら/これらの/問題は/
(and unfortunately, these problems)
二十一世紀にも/継続し/
(to continue during also 21st century,)
あるいは/悲観的な/見方を/すれば/
(or if we look through blue glasses)
さらに/悪くなるという風に/思われます
(seems to be come worse.)

$$precision = \frac{\#\ \ of\ \ correctly\ \ inserted\ \ linefeeds}{\#\ \ of\ \ inserted\ \ linefeeds}$$

$$F\text{-}measure = \frac{2 * recall * precison}{recall + precision}$$

Figure 10 Example of the correct data



(original sentence)

それから二番目に先程伊藤さんからもお話ございましたように今年は終戦五十年ということで特別の年でございますのでそれに関することを若干話させて頂きたいと思います

(Second, as Mr. Ito has just talked about that this year marks the fiftieth anniversary of the end of World War Two. As this is a special year, I'd like to address that event briefly.)

それから現在我々が住んでおります冷戦後の世界というものはどういうものかという点につきまして私の考えを述べさせて頂きたいと思います

(Then, we are now living in the world after the Cold War and then, I'd like to share my opinion on what it looks like.)

そして最後に二十一世紀の日本外交なんて言ってしまって若干後悔しているんですが二十一世紀といっても五十年百年後というところは予測が不可能でございますが二十一世紀の初めの方はどうなるのだろうかとまたその二十一世紀に入って我々としてはどうすべきかということについて私なりの考えを話させて頂きたいと思います

(Lastly, I've already said "the Japanese diplomacy during the twenty-first century" and I'm now regretting that a bit. Although I mention "the twenty-first century" in one word, no one can predict what the world will be like after fifty or a hundred years from now. As for the early part of the twenty-first century, what do we predict will happen? And after having entered the twenty-first century, how should we behave? I'd like to express my opinion in answer to these questions.)

(correct data)

それから二番目に
先程伊藤さんからもお話ございましたように
今年は終戦五十年ということで
特別の年でございますので
それに関することを
若干話させて頂きたいと思います
それから現在我々が住んでおります
冷戦後の世界というものは
どういうものかという点につきまして
私の考えを述べさせて頂きたいと思います
そして最後に
二十一世紀の日本外交なんて言ってしまって
若干後悔しているんですが
二十一世紀といっても
五十年百年後というところは
予測が不可能でございますが
二十一世紀の初めの方は
どうなるのだろうかと
またその二十一世紀に入って
我々としては
どうすべきかということについて
私なりの考えを話させて頂きたいと思います

(Second,)
(as Mr. Ito has just talked about that)
(this year marks the fiftieth anniversary of the end of World War Two.)
(As this is a special year,)
(that event,)
(I'd like to address it briefly.)
(Then, we are now living in)
(the world after the Cold War and)
(what it looks like)
(then, I'd like to share my opinion on.)
(Lastly,)
(I've already said "the Japanese diplomacy during the twenty-first century" and)
(I'm now regretting that a bit.)
(Although I mention "the twenty-first century" in one word,)
(like after fifty or a hundred years from now)
(no one can predict what the world will be.)
(As for the early part of the twenty-first century,)
(what do we predict will happen?)
(And after having entered the twenty-first century,)
(we)
(how should we behave?)
(I'd like to express my opinion in answer to these questions.)

Three persons decided the correct data through the consultation. Figure 10 shows an example of the correct data. In the test data, there are 5,497 linefeed points.

5.2 Experimental results

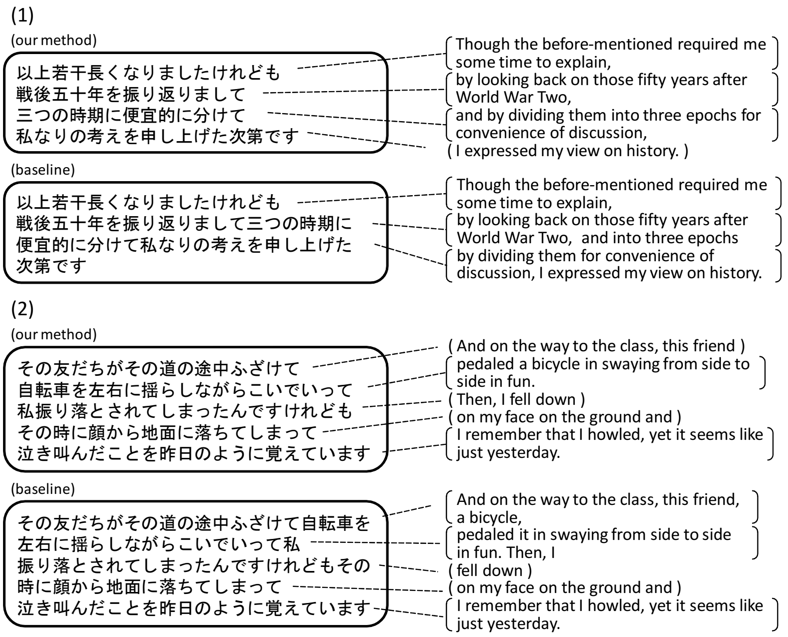
Table 4 shows the experimental results. The recall and precision were 80.2% and 67.4% respectively, and we confirmed that our method had higher performance than the baseline method. Figure 11 shows the result of linefeeds insertion into a spoken monologue sentence in the test data

(1) 以上若干長くなりましたけれども戦後五十年を振り返りまして三つの時期に便宜的に分けて私なりの考えを申し上げた次第です
(Though the before-mentioned required me some time to explain, by looking back on those fifty years after World War Two and by dividing them into three epochs for convenience of discussion, I expressed my view on history.)

Table 4 Experimental results

	recall	*precision*	*F-measure*
our method	80.2% (4,407/5,497)	67.4% (4,407/6,541)	73.2
baseline	27.5% (1,510/5,497)	34.5% (1,510/4,376)	30.6

Figure 11 Examples of linefeed insertion by our method and baseline



以上若干長くなりましたけれども
戦後五十年を振り返りまして
三つの時期に便宜的に分けて
私なりの考えを申し上げた次第です

Though the before-mentioned required me some time to explain,
by looking back on those fifty years after World War Two,
and by dividing them into three epochs for convenience of discussion,
(I expressed my view on history.)

以上若干長くなりましたけれども
戦後五十年を振り返りまして三つの時期に
便宜的に分けて私なりの考えを申し上げた
次第です

Though the before-mentioned required me some time to explain,
by looking back on those fifty years after World War Two, and into three epochs
by dividing them for convenience of discussion, I expressed my view on history.

その友だちがその道の途中ふざけて
自転車を左右に揺らしながらこいでいって
私振り落とされてしまったんですけれども
その時に顔から地面に落ちてしまって
泣き叫んだことを昨日のように覚えています

(And on the way to the class, this friend)
pedaled a bicycle in swaying from side to side in fun.
(Then, I fell down)
(on my face on the ground and)
I remember that I howled, yet it seems like just yesterday.

その友だちがその道の途中ふざけて自転車を
左右に揺らしながらこいでいって私
振り落とされてしまったんですけれどもその
時に顔から地面に落ちてしまって
泣き叫んだことを昨日のように覚えています

And on the way to the class, this friend, a bicycle,
pedaled it in swaying from side to side in fun. Then, I
(fell down)
(on my face on the ground and)
I remember that I howled, yet it seems like just yesterday.

(2) その友だちがその道の途中ふざけて自転車を左右に揺らしながらこいでいって私振り落とされてしまったんですけれどもその時に顔から地面に落ちてしまって泣き叫んだことを昨日のように覚えています
(And on the way to the class, this friend pedaled a bicycle in swaying from side to side in fun. Then, I fell down on my face on the ground and I remember that I howled, yet it seems like jut yesterday.)

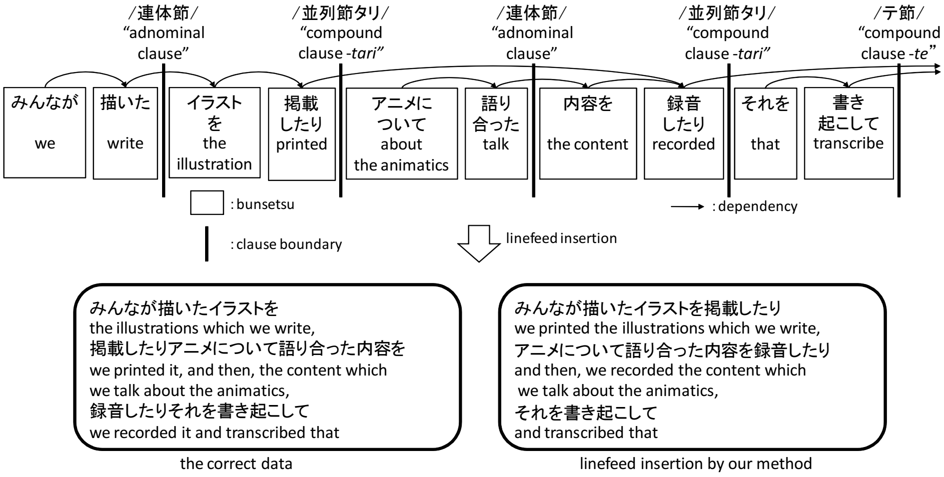
Compared with the baseline, our method could insert linefeeds into proper points. As mentioned above, we confirmed the effectiveness of our method.

Here, we discuss the causes of incorrect linefeed insertion occured in our method. First, Table 5 shows the causes of unnecessary linefeed insertion into the incorrect points. The largest cause is the linefeed insertion based on pauses. There were a lot of cases that the length of a line becomes too short by inserting a linefeed into the bunsetsu boundary in which a pause exists. We need to establish detailed rules based on not only the existence of a pause but also the number of characters in a line and the morphological information such as the part-of-speech of a particle.

Table 5 Causes of incorrect linefeed insertion

causes	#
insertion into the strong clause boundary	125
insertion into the weak clause boundary	214
insertion based on adnominal clause	212
insertion based on clause boundary labeled "topicalized element *-wa*"	202
insertion based on pauses	891
insertion based on dependency distance	91
insertion based on the number of characters of line	383
others	16
total	2,134

Figure 12 Example of the clause boundary which did not appear in the analysis data


/連体節/ "adnominal clause"
/並列節タリ/ "compound clause *-tari*"
/連体節/ "adnominal clause"
/並列節タリ/ "compound clause *-tari*"
/テ節/ "compound clause *-te*"
みんなが we
描いた write
イラストを the illustration
掲載したり printed
アニメについて about the animatics
語り合った talk
内容を the content
録音したり recorded
それを that
書き起こして transcribe
: bunsetsu
: dependency
: clause boundary
linefeed insertion

みんなが描いたイラストを
the illustrations which we write,
掲載したりアニメについて語り合った内容を
we printed it, and then, the content which
we talk about the animatics,
録音したりそれを書き起こして
we recorded it and transcribed that

the correct data

みんなが描いたイラストを掲載したり
we printed the illustrations which we write,
アニメについて語り合った内容を録音したり
and then, we recorded the content which
we talk about the animatics,
それを書き起こして
and transcribed that

linefeed insertion by our method

Second, we discuss the case that our method could not insert linefeeds into the correct points. One of the reasons is the existence of the clause boundaries which did not appear in the analysis data. Figure 12 shows an example. In this example, the clause boundary labeled "compound clause *-tari* " is a linefeed point. However, there did not exist the clause boundary labeled "compound clause *-tari*" in the learning data. Therefore, the linefeed was incorrectly inserted into the end boundary of a modifier bunsetsu of the adnominal clause. Since the coverage of the current rules are not enough, we need to increase the size of the learning data.

6 Conclusions

This paper proposed a method for inserting linefeeds into Japanese monologue sentences to support the understanding of monologue speech by the deaf persons, elderly persons or foreigners. Our method can insert linefeeds so that captions become easy to be read by applying the rules which are established based on the emerging pattern of morphemes, dependencies, clause boundaries, pauses, fillers and so on. An experiment on inserting linefeeds by using a monologue corpus showed the recall and precision was 80.2% and 67.4%, respectively, and we confirmed the effectiveness of our method.

In applying the linefeed insertion technique to practical real-time captioning, we have to consider not only the readability but also the simultaneity. Since the input of our method is a sentence which tends to be long in spoken monologue, in the future, we will develop a more simultaneous technique of which the input is shorter than a sentence.

Acknowledgements

This research was partially supported by the Grant-in-Aid for Scientific Research (B) and Young Scientists (B) of JSPS and by The Telecommunications Advancement Foundation.

References

Boulianne, G., Beaumont, J.-F., Boisvert, M., Brousseau, J., Cardinal, P., Chapdelaine, C., Comeau, M., Ouellet, P. and Osterrath, F. (2006) 'Computer-assisted closed-captioning of live TV broadcasts in French', *Proceedings of the 9th International Conference on Spoken Language Processing (Interspeech 2006-ICSLP)*, 17–21 September, Pittsburgh, USA, pp.273–276.

Daelemans, W., Hothker, A. and Sang, E. T. K. (2004) 'Automatic sentence simplification for subtitling in Dutch and English', *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, 26–28 May, Lisbon, Portugal, pp.1045–1048.

Holter, T., Harborg, E., Johnsen, M. H. and Svendsen, T. (2000) 'ASR-based subtitling of live TV-programs for the hearing impaired', *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP 2000)*, 16–20 October, Beijing, China, pp.570–573.

Imai, T., Sato, S., Kobayashi, A., Onoe, K. and Homma, S. (2006) 'Online speech detection and dual-gender speech recognition for captioning broadcast news', *Proceedings of the 9th International Conference on Spoken Language Processing (Interspeech 2006-ICSLP)*, 17–21 September, Pittsburgh, USA, pp.1602–1605.

Iwata, K., Mitome, Y. and Watanabe, T. (1990) 'Pause rule for Japanese text-to-speech conversion using pause insertion probability', *Proceedings of the 1st International Conference on Spoken Language Processing (ICSLP 90)*, 18–22 November, Kobe, Japan, pp.837–840.

Kashioka, H., and Maruyama, T. (2004) 'Segmentation of semantic units in Japanese monologues', *Proceedings of the International Conference on Speech and Language*

Technology (ICSLT 2004 and Oriental-COCOSDA 2004), 17–19 November, New Delhi, India, pp.87–92.

Kudo, T. and Matsumoto, Y. (2002) 'Japanese Dependency Analysis using Cascaded Chunking', *Proceedings of the Conference on Natural Language Learning (CoNLL 2002)*, 31 August and 1 September, Taipei, Taiwan, pp.63–69.

Kurohashi, S. and Nagao, M. (1997) 'Building a Japanese parsed corpus while improving the parsing system', *Proceedings of the 3rd Natural Language Processing Pacific Rim Symposium (NLPRS 97)*, 2–4 December, Phuket, Thailand, pp.451–456.

Maekawa, K., Koiso, H., Furui, S. and Isahara, H. (2000) 'Spontaneous speech corpus of Japanese', *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, 29–31 May, Las Palmas, Spain, pp.947–952.

Matsubara, S., Takagi, A., Kawaguchi, N. and Inagaki, Y. (2002) 'Bilingual spoken monologue corpus for simultaneous machine interpretation research', *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, 29–31 May, Las Palmas, Spain, pp.153–159.

Matsumoto, Y., Kitauchi, A., Yamashita, T. and Hirano, Y. (1999) *Japanese morphological analysis system ChaSen version 2.0 manual*, Nara Institute of Science and Technology, Nara.

Matsumoto, Y. and Asahara, M. (2001) *IPADIC user's manual version 2.2.4*, Nara Institute of Science and Technology, Nara. (in Japanese)

Monma, T., Sawamura, E., Fukushima, T., Maruyama, I., Ehara, T. and Shirai, K. (2003) 'Automatic closed-caption production system on TV programs for hearing-impaired people', *Journal of Systems and Computers in Japan*, Vol. 34(13), pp.71–82.

Munteanu, C., Penn, G. and Baecker. R. (2007) 'Web-based language modelling for automatic lecture transcription', *Proceedings of the 8th Annual Conference of the International Speech Communication Association (Interspeech 2007)*, 28–31 August, Antwerp, Belgium, pp.2353–2356.

Nakano, S., Makihara, T., Kanazawa, T., Nakano, Y., Arai, T., Kuroki, H., Ino, S. and Ifukube, T. (2007) 'Issues of real-time captioning systems using speech recognition technology for deaf and hard-of-hearing persons – Influences of properties of spoken language for sentence-comprehension –', *Journal of IEICE Transactions on information and systems*, Vol. J90-D, No. 3, pp.808–814. (in Japanese)

Ohno, T., Matsubara, S., Kashioka, H., Kato, N. and Inagaki, Y. (2006) 'A syntactically annotated corpus of Japanese spoken monologue', *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, 24–26 May, Genoa, Italy, pp.1590–1595.

Ohno, T., Murata, M. and Matsubara, S. (2009) 'Linefeed insertion into Japanese spoken monologue for captioning', *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2009)*, 2–7 August, Suntec, Singapore, pp.531–539.

Saiko, M., Takanashi, K. and Kawahara, T. (2005) 'Cascaded chunking of spontaneous Japanese using bunsetsu dependency and pause information', *IPSJ SIG Technical Reports*, Vol. 2005, No. 127, pp.247–252. (in Japanese)

Saraclar, M., Riley, M., Bocchieri, E. and Goffin. V. (2002) 'Towards automatic closed captioning: Low latency real time broadcast news transcription', *Proceedings of the 7th International Conference on Spoken Language Processing (Interspeech 2002-ICSLP)*, 16–20, September, Denver, USA, pp.1741–1744.

Xue, J., Hu. R., and Zhao. Y. (2006) 'New improvements in decoding speed and latency for automatic captioning', *Proceedings of the 9th International Conference on Spoken Language Processing (Interspeech 2006-ICSLP)*, Pittsburgh, USA, 17–21 September, pp.1630–1633.

Notes

[1] *Bunsetsu* is a linguistic unit in Japanese that roughly corresponds to a basic phrase in English. A bunsetsu consists of one independent word and zero or more ancillary words. A *dependency* is a modification relation in which a *modifier bunsetsu* depends on a *modified bunsetsu*. That is, the modifier bunsetsu and the modified bunsetsu work as modifier and modifyee, respectively.

[2] We used ChaSen (Matsumoto et al., 1999) as morpholigical analyzer, CBAP (Kashioka and Maruyama, 2004) as clause boundary analyzer and CaboCha (Kudo and Matsumoto, 2002) with default learning data as dependency parser.

[3] The specification of the parts-of-speech is in accordance with that of IPA parts-of-speech (Matsumoto and Asahara, 2001) in a morphological analyzer called ChaSen (Matsumoto et al., 1999), the rules of the bunsetsu segmentation with those of CSJ (Maekawa et al., 2000), and the dependency grammar is in accordance with that of the Kyoto Text Corpus (Kurohashi and Nagao, 1997)

[4] We used the types of clause boundaries defined by the Clause Boundary Annotation Program (Kashioka and Maruyama, 2004). The type of a clause boundary is decided by the clause of which the end boundary is the clause boundary. There exist 147 types of clause boundaries, and the types of clause boundaries are represented by annotation labels such as "compound clause" and "adnominal clause."