

Text Generation from Keywords

Kiyotaka Uchimoto[†]

Satoshi Sekine[‡]

Hitoshi Isahara[†]

[†]Communications Research Laboratory
2-2-2, Hikari-dai, Seika-cho, Soraku-gun,
Kyoto, 619-0289, Japan
{uchimoto, isahara}@crl.go.jp

[‡]New York University
715 Broadway, 7th floor
New York, NY 10003, USA
sekine@cs.nyu.edu

Abstract

We describe a method for generating sentences from “keywords” or “headwords”. This method consists of two main parts, candidate-text construction and evaluation. The construction part generates text sentences in the form of dependency trees by using complementary information to replace information that is missing because of a “knowledge gap” and other missing function words to generate natural text sentences based on a particular monolingual corpus. The evaluation part consists of a model for generating an appropriate text when given keywords. This model considers not only word n-gram information, but also dependency information between words. Furthermore, it considers both string information and morphological information.

1 Introduction

Text generation is an important technique used for applications like machine translation, summarization, and human/computer dialogue. In recent years, many corpora have become available, and have been used to generate natural surface sentences. For example, corpora have been used to generate sentences for language model estimation in statistical machine translation. In such translation, given a source language text, S , the translated text, T , in the target language that maximizes the probability $P(T|S)$ is selected as the most appropriate translation, T_{best} , which is represented as (Brown et al., 1990)

$$\begin{aligned} T_{best} &= \operatorname{argmax}_T P(T|S) \\ &= \operatorname{argmax}_T (P(S|T) \times P(T)). \end{aligned} \quad (1)$$

In this equation, $P(S|T)$ represents the model used to replace words or phrases in a source language with those in the target language. It is called a *translation model*. $P(T)$ represents a language model that is used to reorder translated words or phrases into a natural order in

the target language. The input of the language model is a “bag of words,” and the goal of the model is basically to reorder the words. At this point, there is an assumption that natural sentences can be generated by merely reordering the words given by a translation model. To give such a complete set of words, however, a translation model needs a large number of bilingual corpora. If we could automatically complement the words needed to generate natural sentences, we would not have to collect the large number of bilingual corpora required by a translation model. In this paper, we assume that the role of the translation model is not to give a complete set of words that can be used to generate natural sentences, but to give a set of headwords or center words that a speaker might want to express, and describe a model that can provide the complementary information needed to generate natural sentences by using a target language corpus when given a set of headwords.

If we denote a set of headwords in a target language as K , we can express Eq. (1) as

$$P(T|S) = P(K|S) \times P(T|K). \quad (2)$$

$P(K|S)$ in this equation represents a model that gives a set of headwords in the target language when given a source-language text sentence. $P(T|K)$ represents a model that generates text sentence T when given a set of headwords, K . We call the model represented by $P(T|K)$ a *text-generation model*. In this paper, we describe a text-generation model and a generation system that uses the model. Given a set of headwords or keywords, our system outputs the text sentence that maximizes $P(T|K)$ as an appropriate text sentence, T_{best} :

$$\begin{aligned} T_{best} &= \operatorname{argmax}_T P(T|K) \\ &= \operatorname{argmax}_T (P(K|T) \times P(T)). \end{aligned} \quad (3)$$

In this equation, we call the model represented by $P(K|T)$ a *keyword-production model*. This equation is equal to Eq. (1) when a source-text sentence is replaced with a set of key-

words. Therefore, this model can be regarded as a model that translates keywords into text sentences. The model represented by $P(T)$ in Eq. (3) is a language model used in statistical machine translation. The n-gram model is the most popular one used as a language model.

We assume that there is one extremely probable ordered set of morphemes and dependencies between words that produce keywords, and we express $P(K|T)$ as

$$\begin{aligned} P(K|T) &\approx P(K, M, D|T) \\ &= P(K|M, D, T) \times P(D|M, T) \times P(M|T). \end{aligned} \quad (4)$$

In this equation, M denotes an ordered set of morphemes and D denotes an ordered set of dependencies in a sentence. $P(K|M, D, T)$ represents a keyword-production model. To estimate the models represented by $P(D|M, T)$ and $P(M|T)$, we use a dependency model and a morpheme model, respectively, for the dependency analysis and morphological analysis.

Statistical machine translation and example-based machine translation require numerous high-quality bilingual corpora. Interlingual machine translation and transfer-based machine translation require a parser with high precision. Therefore, these approaches to translation are not practical if we do not have enough bilingual corpora or a good parser. This is especially so if the source text-sentences are incomplete or have errors like those often found in OCR and speech-recognition output. In these cases, however, if we translate headwords into words in the target language and generate sentences from the translated words by using our method, we should be able to generate natural sentences from which we can grasp the meaning of the source-text sentences.

The text-generation model represented by $P(T|K)$ in Eq. (2) can be applied to various tasks besides machine translation.

- **Sentence-generation support system for people with aphasia:** About 300,000 people are reported to suffer from aphasia in Japan, and 40% of them can select only a few words to describe a picture. If candidate sentences can be generated from these few words, it would help these people communicate with their families and friends.
- **Support system for second language writing:** Beginners writing in second language usually find it easy to produce center words or headwords, but often have difficulty generating complete sentences. If several possible sentences could be generated from those words, it would help begin-

ners communicate with foreigners or study second-language writing.

These are just two examples. We believe that there are many other possible applications.

2 Overview of the Text-Generation System

In this section, we give an overview of our system for generating text sentences from given keywords. As shown in Fig. 1, this system consists of three parts: generation-rule acquisition, candidate-text sentence construction, and evaluation.

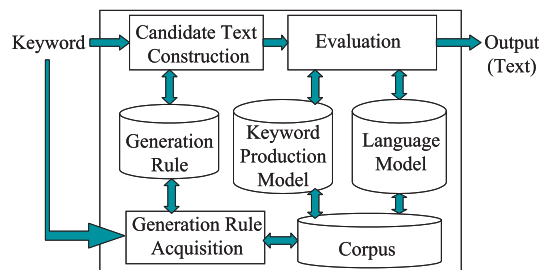


Figure 1: Overview of the text-generation system.

Given keywords, text sentences are generated as follows.

1. During generation-rule acquisition, generation rules for each keyword are automatically acquired.
2. Candidate-text sentences are constructed during candidate-text construction by applying the rules acquired in the first step. Each candidate-text sentence is represented by a graph or dependency tree.
3. Candidate-text sentences are ranked according to their scores assigned during evaluation. The scores are calculated as a probability estimated by using a keyword-production model and a language model that are trained with a corpus.
4. The candidate-text sentence that maximizes the score or the candidate-text sentences whose scores are over a threshold are selected as output. The system can also output candidate-text sentences that are ranked within the top N sentences.

In this paper, we assume that the target language is Japanese. We define a keyword as the headword of a *bunsetsu*. A *bunsetsu* is a phrasal unit that usually consists of several content and function words. We define the headword of a *bunsetsu* as the rightmost content word in the *bunsetsu*, and we define a content word as a

word whose part-of-speech is a verb, adjective, noun, demonstrative, adverb, conjunction, attribute, interjection, or undefined word. We define the other words as function words. We define formal nouns and auxiliary verbs “SURU (do)” and “NARU (become)” as function words, except when there are no other content words in the same *bunsetsu*. Part-of-speech categories follow those in the Kyoto University text corpus (Version 3.0) (Kurohashi and Nagao, 1997), a tagged corpus of the *Mainichi* newspaper.

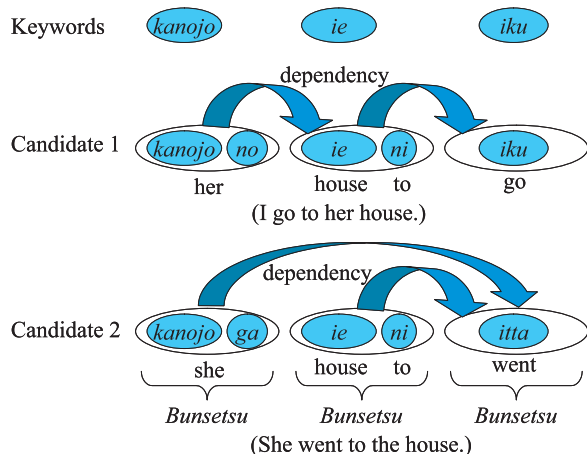


Figure 2: Example of text generated from keywords.

For example, given the set of keywords “*kanojo* (she),” “*ie* (house),” and “*iku* (go),” as shown in Fig. 2, our system retrieves sentences including each word, and extracts each *bunsetsu* that includes each word as a headword of the *bunsetsu*. If there is no tagged corpus such as the Kyoto University text corpus, each *bunsetsu* can be extracted by using a morphological-analysis system and a dependency-analysis system such as JUMAN (Kurohashi and Nagao, 1999) and KNP (Kurohashi, 1998). Our system then acquires generation rules as follows.

- “*kanojo* (she)→*kanojo* (she) *no* (of)”
- “*kanojo* (she)→*kanojo* (she) *ga*”
- “*ie* (house)→*ie* (house) *ni* (to)”
- “*iku* (go)→*iku* (go)”
- “*iku* (go)→*itta* (went)”

The system next generates candidate *bunsetsus* for each keyword and candidate-text sentences in the form of dependency trees, such as “Candidate 1” and “Candidate 2” in Fig. 2, with the assumption that there are dependencies between keywords. Finally, the candidate-text sentences are ranked by their scores, calculated by a text-generation model, and transformed into surface sentences.

In this paper, we focus on the keyword-production model represented by Eq. (4) and assume that our system outputs sentences in the form of dependency trees.

3 Candidate-Text Construction

We automatically acquire generation rules from a monolingual target corpus at the time of generating candidate-text sentences. Generation rules are restricted to those that generate *bunsetsus*, and the generated *bunsetsus* must include each input keyword as a headword in the *bunsetsu*. We then generate candidate-text sentences in the form of dependency trees by simply combining the *bunsetsus* generated by the rules. The simple combination of generated *bunsetsus* may produce semantically or grammatically inappropriate candidate-text sentences, but our goal in this work was to generate a variety of text sentences rather than a few fixed expressions with high precision¹.

3.1 Generation-Rule Acquisition

Let us denote a set of keywords as KS and a set of rules, each of which generates a *bunsetsu* when given keyword $k(\in KS)$, as R_k . We then restrict $r_k(\in R_k)$ to those represented as

$$k \rightarrow h_k m^*. \quad (5)$$

In this rule, h_k represents the head morpheme whose word is equal to keyword k ; m^* represents zero, one, or a series of morphemes that are connected to h_k in the same *bunsetsu*. Here, we define a morpheme as consisting of a word and its morphological information or grammatical attribute, such as part-of-speech, and we define a head morpheme as consisting of a headword and its grammatical attribute. By applying these rules, we generate *bunsetsus* from input keywords.

3.2 Construction of Dependency Trees

Given keywords $K = k_1 k_2 \dots k_n$, candidate *bunsetsus* are generated by applying the generation rules described in Section 3.1. Next, by assuming dependency relationships between the *bunsetsus*, candidate dependency trees are constructed. Dependencies between the *bunsetsus* are restricted in that they must have the following characteristics of Japanese dependencies:

¹Note that 83.33% (3,973/4,768) of the headwords in the newspaper articles appearing on January 17, 1995 were found in those appearing from January 1st to 16th. However, only 21.82% (2,295/10,517) of the headword dependencies in the newspaper articles appearing on January 17th were found in those appearing from January 1st to 16th.

- (i) Dependencies are directed from left to right.
- (ii) Dependencies do not cross.
- (iii) All *bunsetsus* except the rightmost one depend on only one other *bunsetsu*.

For example, when three keywords are given and candidate *bunsetsus* including each keyword are generated as b_1 , b_2 , and b_3 , the candidate dependency trees are $(b_1 (b_2 b_3))$ and $((b_1 b_2) b_3)$ if we do not reorder keywords, but 16 trees result if we consider the order of keywords to be arbitrary.

4 Text-Generation Model

We next describe the model represented by Eq. (4); that is, a keyword-production model, a morpheme model that estimates how likely a string is to be a morpheme, and a dependency model. The goal of this model is to select optimal sets of morphemes and dependencies that can generate natural sentences. We implemented these models within an maximum entropy framework (Berger et al., 1996; Ristad, 1997; Ristad, 1998).

4.1 Keyword-Production Models

This section describes five keyword-production models which are represented by $P(K|M, D, T)$ in Eq. (4). In these models, we define the set of headwords whose frequency in the corpus is over a certain threshold as a set of keywords, KS , and we restrict the *bunsetsus* to those generated by the generation rules represented in form (5). We assume that all keywords are independent and that k_i corresponds to word $w_j (1 \leq j \leq m)$ when text is given as a series of words $w_1 \dots w_m$.

1. trigram model

We assume that k_i depends only on the two anterior words w_{j-1} and w_{j-2} .

$$P(K|M, D, T) = \prod_{i=1}^n P(k_i | w_{j-1}, w_{j-2}) \quad (6)$$

2. posterior trigram model

We assume that k_i depends only on the two posterior words w_{j+1} and w_{j+2} .

$$P(K|M, D, T) = \prod_{i=1}^n P(k_i | w_{j+1}, w_{j+2}) \quad (7)$$

3. dependency bigram model

We assume that k_i depends only on the two rightmost words w_l and w_{l-1} in the rightmost *bunsetsu* that modifies the *bunsetsu* including k_i (see Fig. 3).

$$P(K|M, D, T) = \prod_{i=1}^n P(k_i | w_l, w_{l-1}). \quad (8)$$

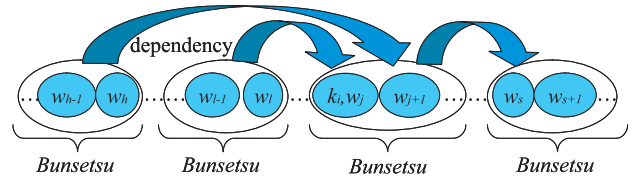


Figure 3: Relationship between keywords and words in *bunsetsus*.

4. posterior dependency bigram model

We assume that k_i depends only on the headword, w_s , and the word on its right, w_{s+1} , in the *bunsetsu* that is modified by the *bunsetsu* including k_i (see Fig. 3).

$$P(K|M, D, T) = \prod_{i=1}^n P(k_i | w_s, w_{s+1}). \quad (9)$$

5. dependency trigram model

We assume that k_i depends only on the two rightmost words w_l and w_{l-1} in the rightmost *bunsetsu* that modifies the *bunsetsu*, and on the two rightmost words w_h and w_{h-1} in the leftmost *bunsetsu* that modifies the *bunsetsu* including k_i (see Fig. 3).

$$P(K|M, D, T) = \prod_{i=1}^n P(k_i | w_l, w_{l-1}, w_h, w_{h-1}). \quad (10)$$

4.2 Morpheme Model

Let us assume that there are l grammatical attributes assigned to morphemes. We call a model that estimates the likelihood that a given string is a morpheme and has the grammatical attribute $j (1 \leq j \leq l)$ a *morpheme model*.

Let us also assume that morphemes in the ordered set of morphemes M depend on the preceding morphemes. We can then represent the probability of M , given text T ; namely, $P(M|T)$ in Eq. (4):

$$P(M|T) = \prod_{i=1}^n P(m_i | m_1^{i-1}, T), \quad (11)$$

where m_i can be one of the grammatical attributes assigned to each morpheme.

4.3 Dependency Model

Let us assume that dependencies $d_i (1 \leq i \leq n)$ in the ordered set of dependencies D are independent. We can then represent $P(D|M, T)$ in Eq. (4) as

$$P(D|M, T) = \prod_{i=1}^n P(d_i | M, T). \quad (12)$$

5 Evaluation

To evaluate our system we made 30 sets of keywords, with three keywords in each set, as shown in Table 1. A human subject selected the sets from headwords that were found ten

Table 1: Input keywords and examples of system output.

Input (Keywords)		Ex. of system output
ロシア	連覇 達成	(ロシアで(連覇を)達成した))
世界	記録 変わる	((世界の記録に)変わる))
課題	実現 挑む	((課題の実現に)挑む))
チーム	ともに 勝つ	(チームで(ともに)勝った))
この	計画 決める	
戦後	生活 支える	((戦後の生活を)支える))
大統領	会見 陳述	((大統領の会見の)陳述))
容疑者	中国 逮捕	(容疑者は(中国で)逮捕された))
日本	五輪 開催	
国	政策 発足	((国の政策が)発足する))
課題	実現 向かう	((課題の実現に)向かう))
社会	働く 女性	((社会に働いている)女性))
首相	政権 奪回	(首相が(政権を)奪回する))
提案	見直し 作る	((提案の見直しを)作る))
政権	発足 決まる	(政権が(発足に)決まった))
香港	クルー 多い	(香港は(クルーも)多いらしい))
大会	出場 勝つ	
正月	旅行 急増	((正月の旅行も)急増している))
民営化	反対 人	((民営化に反対している)人))
昨年	政策 認める	((昨年の政策を)認めるべきだ))
米国	勝つ 明らかだ	((米国に勝てば)明らかになる))
住民	疑い 広がる	((住民の疑いが)広がる))
日本	中国 近い	
外国人	加入 増加	((外国人の加入者が)増加している))
五輪	選手権 獲得	((五輪の選手権を)獲得する))
企業	撤廃 進める	(企業が(撤廃を)進める))
将来	新進党 生まれる	(将来は(新進党が)生まれるだろう))
昨年	記録 上回る	((昨年の記録を)上回る))
強い	チーム 目指す	
よい	仕事 ほしい	

times or more in the newspaper articles on January 1st in the Kyoto University text corpus (Version 3.0) without looking at the articles.

We evaluated each model by the percentage of outputs that were subjectively judged as appropriate by one of the authors. We used two evaluation standards.

- Standard 1: If the dependency tree ranked first is semantically and grammatically appropriate, it is judged as appropriate.
- Standard 2: If there is at least one dependency tree that is ranked within the top ten and is semantically and grammatically appropriate, it is judged as appropriate.

We used headwords that were found five times or more in the newspaper articles appearing from January 1st to 16th in the Kyoto University text corpus and also found in those appearing on January 1st as the set of headwords, *KS*. For headwords that were not in *KS*, we added their major part-of-speech categories to the set. We trained our keyword-production models by using 1,129 sentences (containing 10,201 headwords) from newspaper articles appearing on January 1st. We used a morpheme model and a dependency model identical to those proposed by Uchimoto et al. (Uchimoto et al., 2001; Uchimoto et al., 1999; Uchimoto et al., 2000b). To train the models, we used 8,835 sentences from

newspaper articles appearing from January 1st to 9th in 1995. Generation rules were acquired from newspaper articles appearing from January 1st to 16th. The total number of sentences was 18,435.

First, we evaluated the outputs generated when the rightmost two keywords, such as “連覇 and 達成,” on each line of Table 1 were input. Table 2 shows the results. KM1 through KM5 stand for the five keyword-production models described in Section 4.1, and MM and DM stand for the morpheme and the dependency models, respectively. The symbol + indicates a combination of models. In the models without MM, DM, or both, $P(M|T)$ and $P(D|M, T)$ were assumed to be 1. We carried out additional experiments with models that considered both the anterior and posterior words, such as the combination of KM1 and KM2 or KM3 and KM4. The results were at most 16/30 by standard 1 and 24/30 by standard 1.

Table 2: Results of subjective evaluation.

Model	Standard 1	Standard 2
KM1 (trigram)	13/30	28/30
KM1 + MM	21/30	28/30
KM1 + DM	12/30	28/30
KM1 + MM + DM	26/30	28/30
KM2 (posterior trigram)	6/30	15/30
KM2 + MM	8/30	20/30
KM2 + DM	10/30	20/30
KM2 + MM + DM	9/30	25/30
KM3 (dependency bigram)	13/30	29/30
KM3 + MM	26/30	29/30
KM3 + DM	14/30	28/30
KM3 + MM + DM	27/30	29/30
KM4 (posterior dependency bigram)	10/30	18/30
KM4 + MM	9/30	26/30
KM4 + DM	9/30	22/30
KM4 + MM + DM	13/30	27/30
KM5 (dependency trigram)	12/30	26/30
KM5 + MM	17/30	28/30
KM5 + DM	12/30	27/30
KM5 + MM + DM	26/30	28/30

The models KM1+MM+DM, KM3+MM+DM, and KM5+MM+DM achieved the best results, as shown in Table 2. For models KM1, KM3, and KM5, the results with MM and DM were significantly better than those without MM and DM in the evaluation by standard 1. We believe this was because cases are more tightly connected with verbs than with nouns, so models KM1, KM3, and KM5, which learn the connection between cases and verbs, can better rank the candidate-text sentences that have a natural connection between cases and verbs than other candidates.

Next, we conducted experiments using the 30 sets of keywords shown in Table 1 as inputs. We used two keyword-production models: model KM3+MM+DM, which achieved

the best results in the first experiment, and model KM5+MM+DM, which considers the richest information. We assumed that the input keyword order was appropriate and did not reorder the keywords. The results for both models were the same: 19/30 in the evaluation by standard 1 and 24/30 in the evaluation by standard 2. The right column of Table 1 shows examples of the system output. For example, for the input “将来 (*syourai*, in the future), 新進党 (*shin-shin-tou*, the New Frontier Party), and 生まれる (*umareru*, to be born)”, the dependency tree “(将来は [*syourai_wa*] (新進党が [*shin-shin-tou_ga*] 生まれるだろう [*umareru_darou*]))” (“The New Frontier Party will be born in the future.”) was generated. This output was automatically complemented by the appropriate modality “だろう” (*darou*, will), which agrees with the word “将来” (*syourai*, in the future), as well as by post-positional particles such as “は” (*wa*, case marker) and “が” (*ga*). For the input “外国人 (*gaikoku-jin*, a foreigner), 加入 (*kanyuu*, to join), and 増加 (*zouka*, to increase)”, the dependency tree “((外国人の [*gaikokujin_no*] 加入者が [*kanyuu_sya_ga*] 増加している [*zouka_shite_iru*])” (“Foreigner members are increasing in number.”) was generated. This output was complemented not only by the modality expression “している” (*shite_iru*, the progressive form) and post-positional particles such as “の” (*no*, of) and “が” (*ga*), but also by the suffix “者” (*sha*, person), and a compound noun “加入者” (*kanyuu_sya*, member) was generated naturally. In six cases, though, we did not obtain appropriate outputs because the candidate-text sentences were not appropriately ranked. Improving the back-off ability of the model by using classified words or synonyms as features should enable us to rank sentences more appropriately.

6 Related Work

Many statistical generation methods have been proposed. In this section, we describe the differences between our method and several previous methods.

Japanese words are often followed by post-positional particles, such as “*ga*” and “*wo*”, to indicate the subject and object of a sentence. There are no corresponding words in English. Instead, English words are preceded by articles, “the” and “a,” to distinguish definite and indefinite nouns, and so on, and in this case there are no corresponding words in Japanese. Knight et al. proposed a way to

compensate for missing information caused by a lack of language-dependent knowledge, or a “knowledge gap” (Knight and Hatzivassiloglou, 1995; Langkilde and Knight, 1998a; Langkilde and Knight, 1998b). They use semantic expressions as input, whereas we use keywords. Also, they construct candidate-text sentences or word lattices by applying rules, and apply their language model, an n-gram model, to select the most appropriate surface text. While we cannot use their rules to generate candidate-text sentences when given keywords, we can apply their language model to our system to generate surface-text sentences from candidate-text sentences in the form of dependency trees. We can also apply the formalism proposed by Langkilde (Langkilde, 2000) to express the candidate-text sentences.

Bangalore and Rambow proposed a method to generate candidate-text sentences in the form of trees (Bangalore and Rambow, 2000). They consider dependency information when deriving trees by using XTAG grammar, but they assume that the input contains dependency information. Our system generates candidate-text sentences without relying on dependency information in the input, and our model estimates the dependencies between keywords.

Ratnaparkhi proposed models to generate text from semantic attributes (Ratnaparkhi, 2000). The input of these models is semantic attributes. His models are similar to ours if the semantic attributes are replaced with keywords. However, his models need a training corpus in which certain words are replaced with semantic attributes. Although our model also needs a training corpus, the corpus can be automatically created by using a morphological analyzer and a dependency analyzer, both of which are readily available.

Humphreys et al. proposed using models developed for sentence-structure analysis to rank candidate-text sentences (Humphreys et al., 2001). As well as models developed for sentence-structure analysis, we also use those developed for morphological analysis and found that these models contribute to the generation of appropriate text.

Berger and Lafferty proposed a language model for information retrieval (Berger and Lafferty, 1999). Their concept is similar to that of our model, which can be regarded as a model that translates keywords into text, while their model can be regarded as one that translates query words into documents. However, the purpose of their model is different: their goal is to

retrieve text that already exists while ours is to generate new text.

7 Conclusion

We have described a method for generating sentences from "keywords" or "headwords". This method consists of two main parts, candidate-text construction and evaluation.

1. The construction part generates text sentences in the form of dependency trees by providing complementary information to replace that missing due to a "knowledge gap" and other missing function words, and thus generates natural text sentences based on a particular monolingual corpus.
2. The evaluation part consists of a model for generating an appropriate text sentence when given keywords. This model considers the dependency information between words as well as word n-gram information. Furthermore, the model considers both string and morphological information.

If a language model, such as a word n-gram model, is applied to the generated-text sentences in the form of dependency trees, an appropriate surface-text sentence is generated. The word-order model proposed by Uchimoto et al. can also generate surface text in a natural order (Uchimoto et al., 2000a).

There are several possible directions for our future research. In particular,

- We would like to expand the generation rules. We restricted the generation rules automatically acquired from a corpus to those that generate a *bunsetsu*. To generate a greater variety of candidate-text sentences, we would like to expand the rules that can generate a dependency tree. Expansion would lead to complementing with content words as well as function words. We also would like to prepare default rules or to classify words into several classes when no sentences including the keywords are found in the target corpus.
- Some of the N-best text sentences generated by our system are semantically and grammatically unnatural. To remove such sentences from among the candidate-text sentences, we must enhance our model so that it can consider more information, such as classified words or those in a thesaurus.
- We restricted keywords to the headwords or rightmost content words in the *bunsetsus*.

We would like to expand the definition of keywords to other content words and to synonyms of the keywords.

Acknowledgments

We thank the Mainichi Newspapers for permission to use their data. We also thank Kimiko Ohta, Hiroko Inui, Takehito Utsuro, Manabu Okumura, Akira Ushioda, Jun'ichi Tsujii, Kiyosi Yasuda, and Masahisa Ohta for their beneficial comments during the progress of this work.

References

- S. Bangalore and O. Rambow. 2000. Exploiting a Probabilistic Hierarchical Model for Generation. In *Proceedings of the COLING*, pages 42–48.
- A. Berger and J. Lafferty. 1999. Information Retrieval as Statistical Translation. In *Proceedings of the ACM SIGIR*, pages 222–229.
- A. L. Berger, S. A. Della Pietra, and V. J. Della Pietra. 1996. A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, 22(1):39–71.
- P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin. 1990. A Statistical Approach to Machine Translation. *Computational Linguistics*, 16(2):79–85.
- K. Humphreys, M. Calcagno, and D. Weise. 2001. Reusing a Statistical Language Model for Generation. In *Proceedings of the EWNLG*.
- K. Knight and V. Hatzivassiloglou. 1995. Two-Level, Many-Paths Generation. In *Proceedings of the ACL*, pages 252–260.
- S. Kurohashi and M. Nagao. 1997. Building a Japanese Parsed Corpus while Improving the Parsing System. In *Proceedings of the NLPRS*, pages 451–456.
- S. Kurohashi and M. Nagao. 1999. *Japanese Morphological Analysis System JUMAN Version 3.61*. Department of Informatics, Kyoto University.
- S. Kurohashi. 1998. *Japanese Dependency/Case Structure Analyzer KNP Version 2.0b6*. Department of Informatics, Kyoto University.
- I. Langkilde and K. Knight. 1998a. Generation that Exploits Corpus-Based Statistical Knowledge. In *Proceedings of the COLING-ACL*, pages 704–710.
- I. Langkilde and K. Knight. 1998b. The Practical Value of N-grams in Generation. In *Proceedings of the INLG*.
- I. Langkilde. 2000. Forest-Based Statistical Sentence Generation. In *Proceedings of the NAACL*, pages 170–177.
- A. Ratnaparkhi. 2000. Trainable Methods for Surface Natural Language Generation. In *Proceedings of the NAACL*, pages 194–201.
- E. S. Ristad. 1997. Maximum Entropy Modeling for Natural Language. ACL/EACL Tutorial Program, Madrid.
- E. S. Ristad. 1998. Maximum Entropy Modeling Toolkit, Release 1.6 beta. <http://www.mnemonic.com/software/memt>.
- K. Uchimoto, S. Sekine, and H. Isahara. 1999. Japanese Dependency Structure Analysis Based on Maximum Entropy Models. In *Proceedings of the EACL*, pages 196–203.
- K. Uchimoto, M. Murata, Q. Ma, S. Sekine, and H. Isahara. 2000a. Word Order Acquisition from Corpora. In *Proceedings of the COLING*, pages 871–877.
- K. Uchimoto, M. Murata, S. Sekine, and H. Isahara. 2000b. Dependency Model Using Posterior Context. In *Proceedings of the IWPT*, pages 321–322.
- K. Uchimoto, S. Sekine, and H. Isahara. 2001. The Unknown Word Problem: a Morphological Analysis of Japanese Using Maximum Entropy Aided by a Dictionary. In *Proceedings of the EMNLP*, pages 91–99.