SIMULTANEOUS SPOKEN LANGUAGE TRANSLATION

Koichiro Ryu

Graduate School of Information Science,
Nagoya University,
Furo-cho, Chikusa-ku,
Nagoya, 464-8601, Japan

Shigeki Matsubara

Information Technology Center,
Nagoya University,
Furo-cho, Chikusa-ku,
Nagoya, 464-8601, Japan

ABSTRACT

This paper proposes a method for incrementally translating English spoken language into Japanese. To realize simultaneous translation between languages with different word order, such as English and Japanese, our method utilizes the feature that the word order of a target language is flexible. We implemented a prototype translation system and conducted an experiment with all 578 sentences in the ATIS corpus. The results indicate improvements in comparison to two other methods.

1. INTRODUCTION

Recently, speech-to-speech translation has become one of the important research topics in machine translation. Though some speech translation systems have been developed so far [1, 4], these systems, because of their sentence-by-sentence translation, cannot start to translate a sentence until it has been fully uttered. One effective method of improving the problem is that a translation system begins to translate the words without waiting for the end of the speaker's utterance, much as a simultaneous interpreter does. To realize simultaneous translation between languages with different word order, such as English and Japanese, our method utilizes the feature that the word order of a target language is flexible. In this paper, we describe a prototype translation system. In order to evaluate it, we conducted an experiment with all 578 sentences in the ATIS corpus.

2. JAPANESE GENERATION IN SIMULTANEOUS ENGLISH-JAPANESE TRANSLATION

Let us consider the following English:

(E1) I want to fly from San Francisco to Denver next Monday.

The standard Japanese for (E1) is

(J1) raishu-no (*'next'*) getsuyobi-ni (*'Monday'*) San Francisco-kara (*'from'*) Denver-he (*'to'*) tobi-tai-to omoi-masu (*'want to fly'*).

Input	Output
I	
want to fly	
from	
San Francisco	
to	
Denver	
next Monday	raishu-no (*'next'*) getsuyobi-ni (*'Monday'*) San Francisco-kara (*'from'*) Denver-he (*'to'*) tobi-tai-to omoi-masu (*'want to fly'*)

(a) The output timing of the translation (J1)

Input	Output
I	
want to fly	
from	
San Francisco	San Francisco-kara (*'from'*)
to	
Denver	Denver-he (*'to'*) tobi-tai-to omoi-masu (*'want to fly'*)
next Monday	raishu-no (*'next'*) getsuyobi-ni (*'Monday'*)

(b) The output timing of the translation (J2)

Fig. 1. The output timing of the translation (J1) and (J2)

Fig.1(a) shows the output timing when the translation is generated as incrementally as possible in consideration of the word alignments between (E1) and (J1). There is "raishu-no getsuyobi-ni" (*'next Monday'*) at the beginning of the translation (J1), and there is "next Monday" corresponding to "raishu-no getsuyobi-ni" at the end of the sentence (E1). Thus, the system cannot output "raishu-no getsuyobi-ni" and its following translation until the whole sentence is uttered. Japanese is a language with a relatively flexible word order. Thus, it is possible that a Japanese translation can be accepted even if it keeps the word order of an English sentence. Let us consider the following Japanese:

(J2) San Francisco-kara (*'from'*) Denver-he (*'to'*) tobi-tai-to omoi-masu (*'want to fly'*) raishu-no (*'next'*) getsuyobi-ni (*'Monday'*).

(J2) can be accepted as the translation of the sentence (E1) and still keep the word order as close as possible to the sentence (E1). Fig.1(b) shows the output timing when the translation is generated as incrementally as possible in consideration of the word alignments between (E1) and (J2). The figure demonstrates that a translation system might be able to output "San Francisco -kara (*'from'*)" when "San Francisco" is input and "Denver-he (*'to'*) tobi-tai-to omoi-masu (*'want to fly'*)" when "Denver" is input. If a translation system outputs the sentence (J2) as the translation of the sentence (E1), the system can translate it incrementally.

Table 1. Comparing our method (Y) with two other methods (X, Z)

Method	Quality				Average delay time	Speaker and interpreter utterance time (sec)
	Perfect	Fair	Acceptable	Nonsense		
X	7 (1.2%)	41 (7.1%)	44 (7.6%)	486 (84.1%)	0	4.7
Y	40 (6.9%)	318 (55.0%)	55 (9.5%)	165 (28.5%)	2.79	6.0
Z	-	-	-	-	3.79	6.4

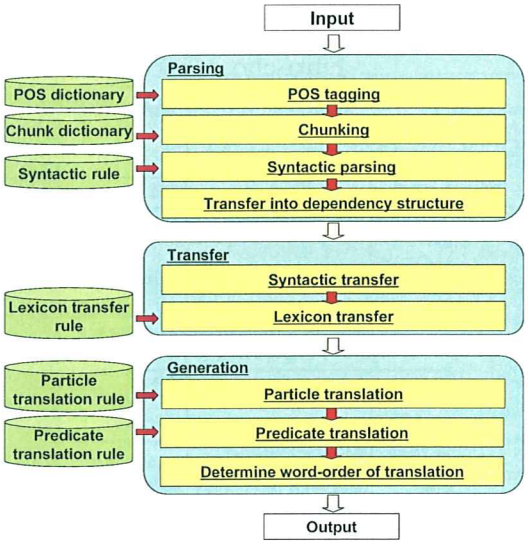

Fig. 2. Configuration of our system

3. SYSTEM CONFIGURATION

Fig.2 shows the configuration of our system [3]. It is composed of three modules: incremental parsing, transfer and generation. In the parsing module the parser determines the English dependency structure for input words incrementally. In the transfer module, structure and lexicon transfer rules transform the English dependency structure into the Japanese case structure. As for the generation module, the system judges whether the translation of each chunk can be output, and if so, outputs the translation of the chunk.

4. EXPERIMENT

To evaluate our method, we conducted a translation experiment was made as follows. We implemented the system in Java language on a 1.0-GHz PentiumM PC with 512 MB of RAM. The experiment used all 578 sentences in the ATIS corpus with a parse tree, in the Penn Treebank [2].

To evaluate the translation quality of our system, each translation result was assigned one of four ranks for translation quality by a human translator. To evaluate the simultaneity of our system, we calculated the average delay time for translating chunks using the following expression:

$$Average\ \ delay\ \ time = \frac{\sum_k d_k}{n}, \qquad (1)$$

where d_k is the virtual elapsed time from inputting the kth chunk until outputting its translated chunk. The virtual elapsed time increases by one unit of time whenever a chunk is input, n is the total number of chunks in all of the test sentences.

We compared the translation results of our method (Y) with two other methods. One method (X) translates the input chunks with no delay time. The other method (Z) translates the input chunks by waiting for the whole sentence to be input, in as consecutive translation. And we virtually compute the delay time and the utterance time. Table 1 shows the estimation results of methods X, Y and Z. Table 1 indicates that our method Y achieved a 55.6% improvement over method X in terms of translation quality and a 1.0 improvement over method Z for the average delay time.

5. CONCLUSION

We have proposed a method for incrementally translating English spoken language into Japanese. Our method utilizes the feature that word order is flexible in Japanese. We implemented a prototype system and conducted an experiment with 578 sentences in the ATIS corpus. We evaluated the translation results of our system in terms of quality and simultaneity.

6. ACKNOLEDGMENTS

The authors would like to thank Prof. Dr. Yasuyoshi Inagaki and Prof. Dr. Toshiki Sakabe for their precious advices.

7. REFERENCES

[1] F. Liu, Y. Gao, L. Gu and M. Picheny, Noise Robustness in Speech to Speech Translation, *IBM Tech Report RC22874*, 2003.

[2] M. P. Marcus, B. Santorini and M. A. Marcinkiewicz, Building a large annotated corpus of English: the Penn Treebank, *Computational Linguistics*, 19(2):310-330, 1993.

[3] K. Ryu, S. Matsubara and Y. Inagaki, Simultaneous English-Japanese Spoken Language Translation Based on Incremental Dependency Parsing and Transfer, *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pp. 683-690.

[4] T. Takezawa, T. Morimoto, Y. Sagisaka, N. Campbell, H. Iida, F. Sugaya, A. Yokoo and S. Yamamoto, A Japanese-to-English Speech Translation System:ATR-MATRIX, *Proceedings of 5th International Conference on Spoken Language Processing*, pp. 957-960, 1998.