

# EXAMPLE-BASED ERROR RECOVERY METHOD FOR SPEECH TRANSLATION: REPAIRING SUB-TREES ACCORDING TO THE SEMANTIC DISTANCE

*Kai Ishikawa, Eiichiro Sumita, and Hitoshi Iida*

ATR Interpreting Telecommunications Research Laboratories  
2-2 Hikaridai, Seika-cho Soraku-gun, Kyoto 619-0288, Japan

## ABSTRACT

In speech translation, recognition errors produced by the speech recognition process can cause parsing and translation errors. Because of this, the development of a robust error handling framework is quite essential to improve the performance of the speech translation system. Previously, a robust translation method was proposed by Wakita, which translates only reliable parts in utterances. In this method, however, the recall of translated parts for a whole utterance is low, and sometimes no translation is output. In this paper, we propose an example-based error recovery method to solve the low recall problem of Wakita's method. The proposed method recovers an unreliable utterance, by repairing the parse-tree of the utterance based on similar example parse-trees in the treebank. A recovered translation is generated from the recovered tree.

## 1. INTRODUCTION

This paper describes a method for recovering speech recognition errors. In speech translation, the translation module needs to swallow utterances including errors caused by the speech recognition process. Accordingly, the development of a robust error handling framework is quite essential to improve the performance of the speech translation system. Y. Wakita et al. proposed a robust translation method [Wakita, 1997], which translates only reliable parts in utterances judged according to their semantic distances calculated for each local sub-tree by using the Constituent Boundary parser (CB-parser) [Furuse, 1996]. In this method, however, the recall of translated parts for a whole utterance is low, and sometimes no translation is output. In this paper, we propose an example-based error recovery method to solve the low recall problem of Wakita's method. The proposed method recovers an unreliable utterance, by repairing the parse-tree of the utterance in the following steps:

- (1) Construct a tree-bank from a text corpus using the CB-parser.
- (2) Obtain the input parse-tree from the recognized utterance using the CB-parser.
- (3) Retrieve example trees phonetically similar to the input sub-tree from the tree-bank.
- (4) Recover the input parse-tree according to the examples.

(5) Generate a translation based on the semantically reliable recovered parse-tree.

## 2. THE ERROR RECOVERY METHOD

### 2.1. The process flow of the proposed recovery method

Figure 1 gives the system overview of a speech translation system with our proposed method. The whole system is composed of three systems, i.e., a speech recognition system (A), a translation system (B, I), and a recovery system (C - H). The translation system is divided into two modules (B, I) and the processes in the recovery system exist between them.

The process flow of the whole system is shown in Fig. 2. We explain the process flow in the following steps (1 - 11), and also the correlation to the modules in Fig. 1 (A - I).

**1 Speech Recognition** Recognize the speech input and then output the text of the recognition result (in A).

**2 Input Parsing** Parse the text input and then output the parse-tree for the input [Furuse, 1996]. The semantic distance for the input is also obtained as a result of the parsing (in B).

**3 Recovery Judgement** Judge the need to recover the input according to its semantic distance. The judgement of the input to be recovered is YES if its semantic distance  $\gamma$  is larger than the threshold  $\Gamma$ , and NO otherwise (in B).

**4 Recover the input?** If the judgement is NO, jump to step 11 (the path from B to I in Fig. 1); skip the recovery steps 5 - 10 (processes in the recovery system: C - H). Otherwise, go to the next step (the path from B to C in Fig. 1).

**5 Sub-tree search** For each sub-tree in the input parse-tree, search for phonetically similar sub-trees (in C) from examples in the tree-bank (D). Phonetic similarity is evaluated as the phonetic distance between the input sub-tree and the example sub-tree, calculated by the edit distance between their phonemes (in E). Filter out the example sub-trees whose phonetic distance  $\delta$  is more than the threshold  $\Delta$ .

**6 Sub-tree obtained?** If no example sub-trees are obtained in the previous step, terminate the recovery process and jump to step 11 (the path from C to I). Otherwise, go to the next step (the path from C to G).

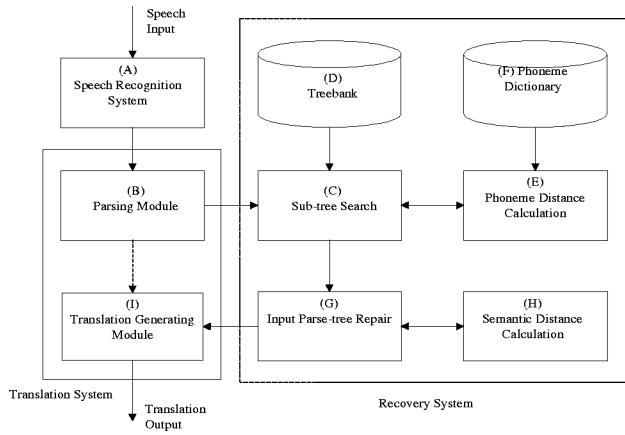
**7 Repair** Repair the input tree by replacing the local part of the input parse tree by the sub-trees obtained in step 5 (in G). The semantic distances for the substituted input parse trees are calculated (in H). We obtain repaired trees for each of the sub-trees obtained in step 5.

**8 Recovery Filtering** Filter out the repaired trees whose semantic distance  $\gamma$  is more than the threshold  $\Gamma$ . Output the tree having the minimum phonetic distance among the repaired trees as the recovered tree.

**9 Recovered tree obtained?** If the recovered tree is not obtained in the previous step (step 8), jump to step 11. Otherwise, go to the next step (the path from G to I).

**10 Recovery Translation** Generate the translation (in I) from the recovered parse tree obtained in step 8.

**11 Original Translation** Generate the translation (in I) from the input parse tree obtained in step 2.



**Figure 1:** System overview of a speech translation system with our error recovery method.

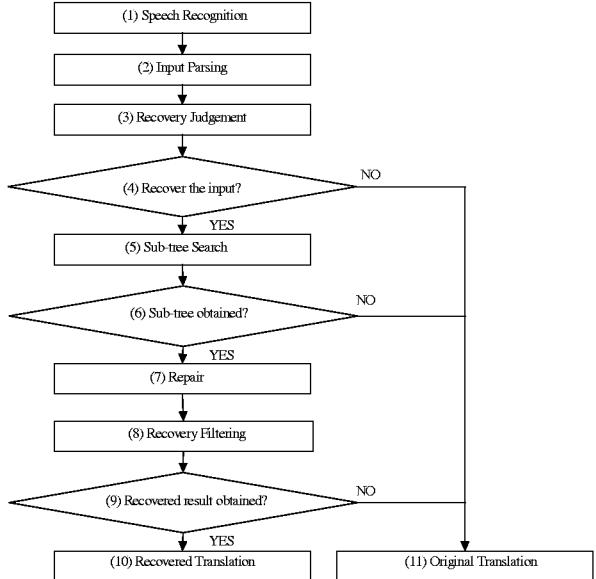
## 2.2. The Parsing and the Semantic distance

The proposed method utilizes the parse-tree and the semantic distance obtained by the CB-parser [Furuse, 1996]. The CB-parser constructs the parse-tree by applying rules in a bottom-up manner. The rules are applied by matching their constituent boundary patterns expressed by either a functional word or a part-of-speech bigram marker. The local semantic distance for the applied rule is calculated according to the semantic distance between the linguistic constituent of the input and the examples defined in the rule for the corresponding constituent.

The semantic distance for the parse-tree is the summation of the local semantic distances. The reliability of the dependency structure in the parse-tree is assured by the small value of its semantic distance. This nature is utilized in the proposed method (steps 3 and 8 in Fig. 2).

The dependency structure of the parse-tree obtained by the CB-parser is robust. Reconstructing the dependency structure

for the repaired parse-tree (step 7 in Fig. 2) is feasible by reflecting the changes of some constituents caused by the repair and by calculating the semantic distance. The tree-bank data is constructed by using the same CB-parser.



**Figure 2:** Process flow of a speech translation system with our error recovery method.

## 2.3. The Sub-tree search according to the phonetic distance

In the sub-tree search process (step 5 in Fig. 2), the phonetic distances between the input sub-tree and the example sub-trees are calculated. In the phonetic distance calculation module (E in Fig. 1), each sub-tree is transformed into a sequence of phonemes by transferring the sequence of terminal words in the sub-tree into the sequence of phonemes by using the phoneme dictionary (F in Fig. 1).

The phonetic similarity is evaluated by the phonetic distance  $\delta$  from the input phonemes to the example phonemes.

The phonetic distance is defined as

$$\delta = (n_{\text{input}} - n_{\text{sub}} - n_{\text{ins}} - n_{\text{del}}) / n_{\text{input}}$$

where the variables are

$n_{\text{input}}$  : the number of total phonemes for the input

$n_{\text{sub}}$  : the number of substituted phonemes in the example

$n_{\text{ins}}$  : the number of inserted phonemes in the example

$n_{\text{del}}$  : the number of deleted phonemes in the example

These variables between the input and the example are obtained by DP-matching of the input and the example phoneme sequences.

### 3. AN EXAMPLE SEQUENCE OF ERROR RECOVERY

We will explain the process flow by showing an example sequence involving Japanese-to-English (JE) translation in Fig. 3. The correspondence of the process to the steps in Fig. 2 is also shown.

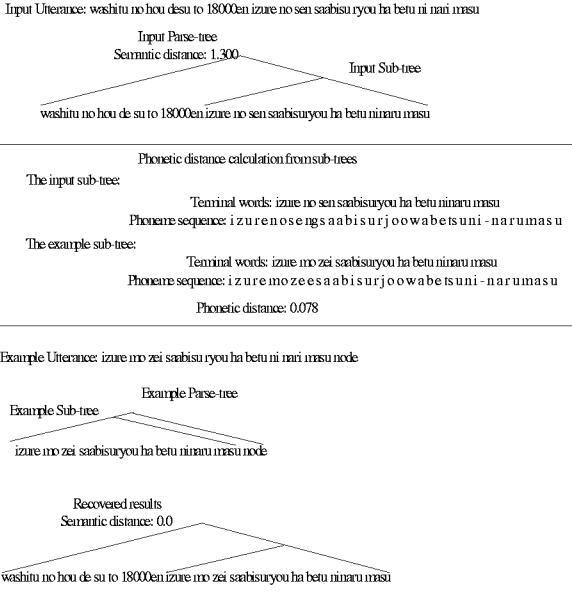


Figure 3: Error recovery example

The utterance, “washitu (a Japanese-style room) no hou desu to (is) 18000 en (18,000 yen) izure no (anyway) sen (match) saabisu (service) ryou (charge) ha (is) betu ni naru masu (not included).”, is the result of the speech recognition (step 1). By parsing the above utterance, we obtain the input parse-tree, with the semantic distance  $\gamma=1.300$  (step 2). In the recovery judgement process (step 3), the input is judged as recovered according to the semantic distance  $\gamma=1.300$ , which is more than the threshold  $\Gamma=0.3$ . In the sub-tree search (step 5), phonetically similar sub-trees to each sub-tree in the input parse tree are retrieved from the tree-bank. For the input sub-tree “izure no sen saabisuryou ha betu ninaru masu”, the phoneme sequence “i z u r e n o s e n g s a a b i s u r j o o w a b e t s u n i - n a r u m a s u” is obtained by using the word-phoneme dictionary. For this input sub-tree, there is a phonetically similar sub-tree “izure mo zei saabisuryou ha betu ninaru masu” in the parse-tree of the example utterance “izure mo zei saabisuryou ha betu ninaru masu node”. The phoneme sequence of this example sub-tree “i z u r e m o z e e s a a b i s u r j o o w a b e t s u n i - n a r u m a s u” is obtained in the same way. By DP-matching of these phoneme sequences, we obtain the phoneme distance of the example sub-tree to the input sub-tree  $\delta=0.078$ . This is smaller than the threshold  $\Delta=0.3$ , so this example sub-tree is filtered through and sent to the repair process. In the repair process (step 7), the input sub-tree of the input parse-tree is substituted with the example sub-tree. By

semantic distance calculation of the substituted tree, we obtain the repaired tree “washitu (a Japanese-style room) no hou desu to (is) 18000en (18,000 yen) izure mo (in either case) zei (tax) saabisu (service) ryou (charge) ha (is) betu ni naru masu (not included).” with the semantic distance  $\gamma=0.0$ . In the recover filtering (step 8), the repaired tree is filtered through according to its semantic distance  $\gamma=0.0$  (this value is less than the threshold  $\Gamma=0.3$ ). Supposing that the repaired tree has the minimum phonetic distance among all candidates, it is sent to the recovery translation process (step 10) and we finally obtain the recovered translation “A Japanese-style room is 18,000yen, and tax and service charge aren’t included in either case.”

Comparing this recovered translation to the original translation of the input utterance i.e., “A Japanese-style room is 18,000yen, and anyway service charge for the match isn’t included.”, we find the recovery to be quite successful.

## 4. EXPERIMENTAL RESULTS

### 4.0. Condition of the experiment

We experimented with our proposed method on Japanese-to-English speech translation. The input 310 utterances were selected from the training set of the translation system. The tree-bank was constructed from 14,111 error-free utterances including the input 310 utterances. We used the threshold of the semantic distance  $\Gamma=0.3$  and the threshold of the phonetic distance  $\Delta=0.3$ . These thresholds were determined according to the results of a preliminary experiment.

Table 1: Recovery Judgement (total of 310 input utterances)

Recovery Judgement	erroneous utterances	correct utterances
Should be recovered 147	141 (67%)	6 (6%)
Need not be recovered 163	71 (33%)	92(94%)
Total 310	212 (100%)	98 (100%)

### 4.1. Recovery judgement

In Table 1, the results of the **recovery judgement** (step 3 in Fig. 2) for both correct utterances and erroneous utterances of the input 310 utterances are shown. For the correct 98 utterances, only 6 (6%) utterances are judged as “should be recovered”. For the erroneous 212 utterances, 141 (67%) utterances are judged as “should be recovered”. These results shows that the utterances are judged as “should be recovered” with a **precision of 94% and a recall of 67%**. The high precision of the recovery judgement is quite meaningful for the performance of the recovery method because it reduces the insecurity of recovering correct utterances. The recall seems rather disappointing, but it is in fact satisfactory in that many of the error utterances under “need not be recovered” are almost correct (except for non-essential errors that do not affect the

translation results). In all, our recover judgement achieves high precision and recall.

## 4.2. Recovery accuracy

Next, we discuss the results of the error recovery. The recovered results are obtained from the 147 utterances (47% of the 310 input utterances) judged as “should be recovered” (in Table 1). An evaluation was performed for the recovered results for the terminal words of the recovered parse trees. To show the effectiveness of the proposed method, we introduced an evaluation measure of recovery for the recovered results shown in Table 2. NIL indicates that the recovered results could not be obtained (step 6 or 9 in Fig. 2).

**Table 2:** Evaluation measure of recovery

<b>Level 5</b>	Exactly the same as the answer
<b>Level 4</b>	Semantically equivalent to the answer
<b>Level 3</b>	Partial recovery/Slightly different from the answer
<b>Level 2</b>	No recovery/Incorrect recovery for erroneous parts of the recognition
<b>Level 1</b>	Incorrect recovery for correct parts of the recognition
<b>NIL</b>	Recovered results are not obtained

The evaluation results are shown in Table 3. The number of utterances for each recovery level defined in Table 2 is shown for two evaluations, i.e., “Experiment 1” and “Experiment 2”.

**Table 3:** Improvement by recovery (total of 147 “should be recovered” utterances)

Rank	Experiment 1	Experiment 2
Level 5	43% (63)	14% (20)
Level 4	5% (8)	6% (9)
level 3	3% (4)	4% (6)
Level 2	1% (2)	2% (3)
Level 1	0% (0)	0% (0)
NIL	48% (70)	74% (109)

“Experiment 1” evaluated the recovery results by using all examples in the tree-bank. In this evaluation, (1) the number of utterances perfectly recovered (in levels 4 and 5) was 71 (48% of the 147 “should be recovered” utterances in Table 1), (2) NIL (not recovered) was 70 (48%), and (3) other problematic recovery cases numbered 6 (4%). Consequently, from the 147 “should be recovered” utterances, **71 erroneous utterances were recovered perfectly without affecting the correct utterances**.

The results shown in “Experiment 1” seem to be successful. But someone may feel that the recovery results obtained by the proposed method are quite trivial by means of the fact that they can be obtained easily, merely by searching for the answer of

the input utterance from the database without any recovery operation.

To show that the proposed method is feasible in non-trivial cases, we also carried out another evaluation; “Experiment 2” in Table 3. In this evaluation, the recovered results were obtained for the same input utterances as in “Experiment 1” **without using the exact answer** of each input utterance. The exact answer of each utterance retrieved from the tree-bank was eliminated and not used in the recovery processes.

In the evaluation of “Experiment 2”, the number of utterances perfectly recovered (in levels 4 and 5) was 29 (20% of the 147 “should be recovered” utterances in Table 1), NIL (not recovered) was 109 (74%), and other problematic recovery causes numbered 9 (6%). Consequently, from the 147 “should be recovered” utterances, **29 erroneous utterances were recovered perfectly without affecting the correct utterances**.

Compared with the results in “Experiment 1”, the number of utterances judged for level 5 is smaller, but the numbers for the other levels (1 - 4) are almost equal. These non-exact recovery results in both evaluations are obtained from the examples not exact to the answer. Therefore, we can conclude that **effective recovery results can be obtained without using the exact answer of the input utterances**.

## 5. CONCLUSION

We showed the feasibility of our error recovery method in Japanese-to-English speech translation. The experimental results demonstrate that the proposed method works only for the erroneous parts of input utterances. The proposed method effectively works even in cases there are no examples exactly the same as the inputs in the tree-bank.

There is a remaining problem. That is about a half of the utterances judged “should be recovered” were not recovered (NIL). This was caused by the strong restriction of the phonetic similarity. To address this, the number of NIL utterances can be reduced by relaxing the phonetic restriction. However, such a relaxation can also cause an increase in the number of inappropriate recovered results filtered out successfully by the strong phonetic restriction.

However, in our proposed method, the semantic distance is also considered independently to assure the reliability of the recovered results. We will investigate the effectiveness of the semantic distance to identify the adequately recovered results.

## REFERENCES

- 1 O. Furuse, H. Iida. 1996. Incremental Translation Utilizing Constituent Boundary Patterns. In Proc. of COLING'96, pp. 412-417.
- 2 Y. Wakita, J. Kawai, H. Iida. 1997. Correct parts extraction from speech recognition results using semantic distance calculation, and its application to speech translation. In Proc. of Spoken Language Translation (ACL '97 Workshop).