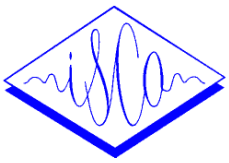

2nd International Conference on
Spoken Language Processing (ICSLP 92)
Banff, Alberta, Canada
October 12-16, 1992

DETECTION OF UNKNOWN WORDS AND AUTOMATIC ESTIMATION OF THEIR TRANSCRIPTIONS IN CONTINUOUS SPEECH RECOGNITION

ITOU Katunobu, HAYAMIZU Satoru†, and TANAKA Hozumi.

Tokyo Institute of Technology, Meguro-ku, Tokyo 152, JAPAN
†Electrotechnical Laboratory, Tsukuba-shi, Ibaraki 305, JAPAN

ABSTRACT

Current continuous speech recognition systems are designed to recognize words within a vocabulary. In order to make speech recognition systems more flexible, convenient and robust, they should be able to process unknown words. This paper introduces a new technique for processing unknown words in a continuous speech recognition system[1]. Two types of processing, one with stochastic language models without any other linguistic knowledge and the other with a dictionary and a grammar, are dynamically controlled to detect and transcribe the unknown words automatically. We tested this method by speaker independent continuous speech recognition experiments using a task with 113 word vocabulary, with bunsetu perplexity 8.2. Preliminary results showed a detection rate for 75% of the unknown words, with a false alarm rate of 11% and a phone recognition rate of 51% for the unknown words detected.

1 INTRODUCTION

Current continuous speech recognition systems are designed to recognize words within a vocabulary. When an unknown word is spoken, a system which cannot deal with unknown words recognizes it as one of the other words in the vocabulary. When this happens, the user cannot know that he/she has uttered an unknown word. He/She assumes that the system simply misrecognized the word. Such a user-interface is very inconvenient and incomprehensible. In order to make speech recognition systems more flexible, convenient and robust, they should be able to process unknown words. Also, for a system that acquires knowledge by natural language dialog with a person, such a technique is indispensable.

A system can respond in various ways for an utterance which contains unknown words. Some examples of ways of responding are as follows.

1. The system recognizes the utterance containing unknown words with the vocabulary alone. If the score of the recognition result is low, the systems reject the utterance.

2. The system recognizes all words except the unknown words in the utterance with the vocabulary, and skips the unknown words.

3. The system recognizes all words except the unknown words in the utterance with the vocabulary, and estimates the transcription of the unknown words.

4. The system recognizes all words except the unknown words in the utterance with the vocabulary, and estimates the meanings of the unknown words.

In order to reject utterances containing the unknown word, methods corresponding to the first of the above have been researched[2]. However, if such a method is used, the system cannot detect an unknown word in the utterance exactly. The user cannot, of course, know whether the rejection is caused by misrecognition or unknown words.

In order to detect unknown words, the method of using a general unknown model was proposed[3, 4]. This technique corresponds to the second of the way of responding listed above, the system can detect unknown words automatically but cannot the estimation of the transcription.

In this paper, we present a new method for processing of unknown words for a continuous speech recognition system[1, 5]. This method not only can detect unknown words but also can estimate their transcriptions automatically.

We use a technique called the phonetic typewriter[6] to estimate and detect unknown words. The phonetic typewriter recognizes the speech input only with a phone HMM and a phone sequence stochastic model. To get high accuracy with the phonetic typewriter requires a lot of computation. In this paper, we also present a method to control the phonetic typewriter to avoid waste of computation.

2 DETECTING UNKNOWN WORDS AND ESTIMATING THEIR TRANSCRIPTIONS

2.1 System Overview

This section describes a system overview of the continuous speech recognition system for processing unknown words. We add the phonetic typewriter to the system in order to process parts of unknown words.

Our speech recognition system consists of two levels of processing for generating the sentence hypotheses: the grammatical level and the lexical level. For the grammatical level, the system uses an LR-parser. When the grammatical level calls the lexical level, it determines the possible grammatical categories for the next word. The lexical level generates word hypotheses by matching an input, and returns the hypothesis to the grammatical level.

10.21437/ICSLP.1992-255

If the input contains an unknown word, the lexical level process fails. Therefore the grammatical level process cannot go on any further and cannot get the correct parse. In order to avoid such a failure, we extend the grammatical level process to deal with input containing unknown words[7] and use the phonetic typewriter[6] as the lexical level process to generate the word hypothesis for unknown words. The system overview is shown in Figure 1.

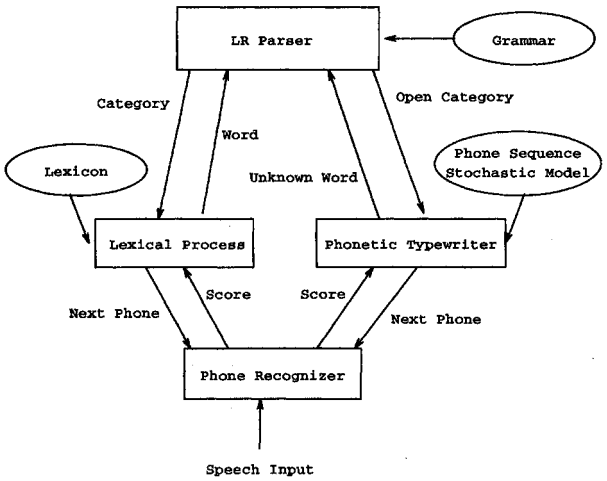

Figure 1: System Overview

In this system, we add the phonetic typewriter to the lexical level. The grammatical level process calls not only the lexical level process but also the phonetic typewriter at the same time. The phonetic typewriter returns the unknown word hypothesis which has the grammatical category that was determined by the grammatical level process and its transcription to the grammatical level process. After generating the unknown word hypothesis using the phonetic typewriter, the grammatical level process deals with the unknown words in the same way as the word had the grammatical category that was determined before generating the unknown word. Therefore, except for the unknown words, the input is dealt with by the lexicon and the grammar, and is able to be recognized correctly.

Unknown words are more likely to appear in specific categories, for example, a proper noun, such as the name of the person. We call such categories open categories. The grammatical level process only calls the phonetic typewriter for open categories.

2.2 The Phonetic Typewriter

For transcribing and detecting unknown words, we must do the phonetic recognition somehow. For that purpose, we use a technique called the phonetic typewriter[6].

The simplest way of doing phonetic recognition is to recognize the input with a full-branching grammar (no grammatical constraint), where each phone can be followed by any other phone with equal probability. However, this requires a lot of computation and the accuracy of phone model is not high enough. Therefore, we use the phone sequence stochastic model as the subword linguistical constraint. The phonetic typewriter recognizes the input with a full-branching grammar and a phone trigram model. The scores of the subword hypotheses are as follows.

$$P_{total} = \mathbf{P}_{hmm} + w_{gram}\mathbf{P}_{gram} \quad (1)$$

$$\mathbf{P}_{hmm} = \frac{\sum_{t=1}^{N_{frame}} \log {P_{hmm}}^{t}}{N_{frame}} \quad (2)$$

$$\mathbf{P}_{gram} = \frac{\sum_{p=1}^{N_{phone}} \log {P_{gram}}^{p}}{N_{phone}} \quad (3)$$

Where, w_{gram} is the weight for phone trigram, N_{frame} is the number of the frame, N_{phone} is the number of the phone, ${P_{hmm}}^{t}$ is the score of the HMM of the frame t, and ${P_{gram}}^{p}$ is the score of the phone trigram of the phone p. The system prunes the hypotheses with the scores under the threshold. The phone N-gram model avoids rare or impossible phone sequences, reduces computation and get high accuracy of the phonetic recognition.

2.3 Control the Phonetic Typewriter

In order to reduce computation, we prune the hypotheses in the phonetic typewriter. However, without any control, the grammatical level process almost always calls the phonetic typewriter. This requires the same large amount of very much computation as is to recognize the whole utterance as phonetic recognition. To get high efficiency and accuracy, we should reduce wasteful computation for the phonetic typewriter as possible. We control the phonetic typewriter by using the score of the lexical level process at each frame as a guide.

Without the phonetic typewriter, the value of P_{hmm} changes in progress of the recognition in Figure 2.

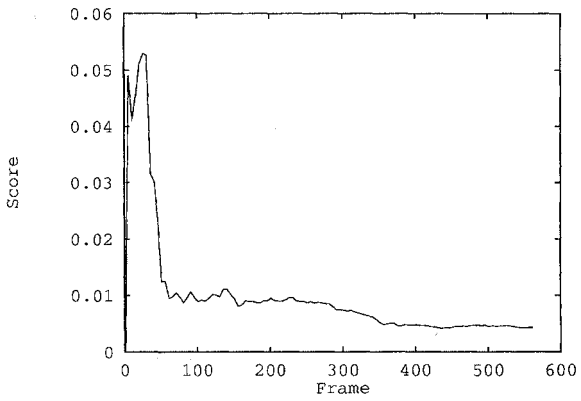

Figure 2: Transition of the score during recognition

In the case of the same utterance containing an unknown word, P_{hmm} changes as shown in Figure 3 by the dotted line. However, if we use the phonetic typewriter to recognize this utterance, P_{hmm} changes as shown by the solid line.

This utterance contains the unknown word around 200 frame. At the beginning of the unknown word, the score is decreasing

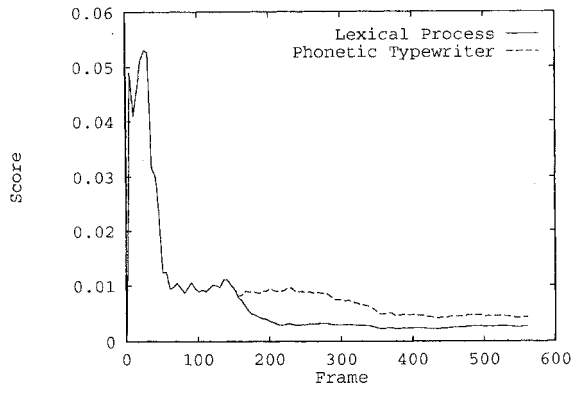

Figure 3: Transition of the score during recognition of an utterance which contains unknown word by the phonetic typewriter and the lexical level process

rapidly. After part of the unknown word, the score becomes constant.

The score of the lexical level process is similar to the score of the phonetic typewriter. Only during part of the unknown word are the two scores are difference. From this observation, we find that the difference between the two scores is greater for part of the unknown word than it is for the rest. Therefore, we control the phonetic typewriter using this difference as a guide. When the difference is smaller, we consider the system is processing the word within the lexicon and the sentence is accepted by the grammar at that time. In this way, the system narrows the threshold for the phonetic typewriter, and the required computation becomes less. On the other hand, when the difference is great, we assume that the system is processing an unknown words. The system then widens the threshold. The required computation becomes greater, and the accuracy of the transcription becomes higher.

3 EXPERIMENTS AND RESULTS

The training data used here includes 1542 words each by 5 speakers and 150 sentences each by 2 speakers.

The phone HMMs are discrete models and have 4 states, 3 loops, and left-to-right structure. They are trained using a forward-backward algorithm. We use context independent model (the number of models was 43.).

We used a phone trigram model as a phone sequence stochastic model. The trigram tables were made from text databases. These data consisted of the newspaper articles, keyboard dialogues, and transcriptions of telephone dialogues. The databases contained 11207 sentences and 859311 phones.

We used a grammar which was defined by a set of 11 templates. An example of such a template is as follows.

$$< \textit{pronoun (place)} > < \textit{case (place)} > < \textit{book (noun)} >$$
$$< \textit{case (agent)} > < \textit{number of book} > < \textit{exist (polite)} >$$

Italicized words bracketed by angle brackets are nonterminal symbols. The size of the lexicon is 113. The bunsetu (a short phrase such as $< \textit{pronoun (place)} > < \textit{case (place)} >$) perplexity of the grammar is 8.2.

The test sentences consisted of 11 sentences each from ten speakers. The texts and the speakers in the test sentences were not included in the training data. Therefore the experiments involved speaker and vocabulary independent recognition.

All the test sentences were analyzed within the grammar and the vocabulary. We conducted the experiments under following condition. We defined all the nominal categories as the open category. In the grammar, 10 templates contained a nominal category. In order to make the utterances contains unknown words, we needed to remove some word from the vocabulary. First, we removed the word /h o N/ (book) from the vocabulary. 5 sentences of the test sentences had one occurrence of unknown word each per a speaker. In the second experiment, we removed 5 words from the original vocabulary. The 5 words belonged to 5 different nominal categories. In this experiment also, 5 sentences of the sentences had one occurrence of unknown word each per a speaker. The total 220 test sentences had 110 occurrences of unknown words.

The detection rate and the estimation rate are shown in Table 1. "det" means the correct detection rate as a percentage of number of unknown words. In this case, the removed word was correctly detected as a unknown word of the same category and the rest of the sentence was correctly recognized. "est" means the exact estimation rate as a percentage of number of unknown words. In this case, the estimated transcription of the removed word was correct and the rest of the sentence was correctly recognized. "c-det" means close detection rate. Here, the removed word was correctly detected as an unknown word of the same category, but the rest of the sentence was misrecognized. "c-est" means close estimation rate. Here, the transcription of the removed word was correct, but the rest of the sentence was misrecognized. "t-det" means total detection rate. The total detection rate is sum of "det" and "c-det". "t-est" means total estimation rate. The total estimation rate is sum of "est" and "c-est". "fal" means false alarm rate and is the ratio of the number of false alarms to the total number of test sentences expressed as a percentage. A false alarm is a unknown word detected where there was no unknown words in the utterance.

det	est	c-det	c-est	t-det	t-est	fal
68	13	7	1	75	14	11

Table 1: Detection and estimation results

The total detection rate was 75% and the total estimation rate was 14%.

The examples which the system couldn't estimate correctly the transcription are shown in Table 2. To calculate the phone accuracy of the words which were detected correctly, we first align the estimated transcription against the correct transcription using the dynamic programming algorithm. Then, we compute the number of occurrences of phone substitutions, insertions, and deletions. Finally, the phone accuracy rate is computed as follows.

$$rate = \frac{Phone - Subs - Dels - Ins}{Phone} \quad (4)$$

Where,

correct	estimated
/hy o- sh i/	/ky o sh i N/
	/o sh i N/
/n e d a N/	/g e d a N/
	/m u d a N/
/ch o sh a/	/ch o sh i o/
	/ky o sh a i/
/k o t o/	/k o t o u/
	/g o u t o u/
/sh a sh i N/	/s a sh i N/
	/s a s u g i N/
/h o N/	/h o o N/
	/h o o-/

Table 2: Examples of the failed transcription

Phone : the number of the phones of the correct transcription
Subs : the number of substitutions
Dels : the number of deletions
Ins : the number of insertions

In the experiments, The phone accuracy rate is 51%.

We also calculated the detection accuracy. We defined the detection accuracy as follows.

$$rate = \frac{U + R}{N} \tag{5}$$

Where,

U : the number of the correctly detected sentences (as defined above) contain unknown words
R : the number of the correctly recognized sentences without unknown words
N : the number of the test sentences

That is, on the sentences contain unknown words, we consider correct detect as success, and on the sentences without unknown words, we consider correct recognition as success. We ran other experiments on the same test sentences. First, we ran a experiment to recognize the test sentences with the original lexicon. Second, we ran a experiment to recognize the test sentences with the deleted lexicon without processing unknown words. We show the results of the experiments in Table 3. To compare with the system without unknown word processing, the system with unknown word processing can increase the number of the sentences which to deal with.

4 Conclusion

In this paper, we have presented the new technique for processing unknown words in a continuous speech recognition system. Using this method, the system can not only detect the unknown word but also estimate the its transcription automatically without repronounciating by the user. In recognition experiments, we demonstrated that the system using the method detected 75% of unknown words and had 11% false alarm rate. The phone recognition accuracy of the correct detected unknown words was 51%.

experiment	detection accuracy
using the original lexicon	85.5
using the deleted lexicon without unknown word processing	46.4
using the deleted lexicon with unknown word processing	66.4

Table 3: Detection accuracy results for with or without unknown processing

Acknowledgement

The authors sincerely wish to express their thanks to the members of Tanaka Laboratory of TIT and to those of Speech Processing Section of ETL. The authors thank to NTT Communication and Information Processing Laboratories and ATR Interpreting Telephony Research Laboratories to permit to use the text database for training the phone trigram model. The "ASJ Continuous Speech Corpus for Research" was used for the training data for phone HMM models.

References

[1] K. Itou, S. Hayamizu, and H. Tanaka. Processing unknown words in continuous speech recognition. *IEICE Tech. Rep.*, pages 41–48, 1991. in Japanese.

[2] S. Tsukada, T. Watanabe, and K. Yoshida. Recognition likelihood normalization for unknown word detection and rejection. In *Proc. ASJ Spring Meeting*, pages 203–204. ASJ, 1991. in Japanese.

[3] A. Asadi, R. Schwartz, and J. Makhoul. Automatic detection of new words in a large vocabulary continuous speech recognition system. In *Proc. ICASSP-90*, pages 125–128, 1990.

[4] A. Asadi, R. Schwartz, and J. Makhoul. Automatic modeling for adding new words to a large-vocabulary continuous speech recognition system. In *Proc. ICASSP-91*, pages 305–308, 1991.

[5] K. Itou, S. Hayamizu, and H. Tanaka. Continuous speech recognition by context-dependent phonetic HMM and an efficient algorithm for finding N-best sentence hypotheses. In *Proc. ICASSP-92*, pages I-21–I-24, 1992.

[6] T. Kawabata, T. Hanazawa, K. Itou, and K. Shikano. Japanese phonetic typewriter using HMM phone recognition and stochastic phone-sequence modeling. *IEICE Transactions*, E 74(7):1783–1787, 1991.

[7] Y. Horiuchi, K. Itou,. and H. Tanaka. A parsing system using generalized lr argorithm for japanese sentence containing undefined words. In *IPSJ Spring Meeting*, pages 325–326, 1990. in Japanese.