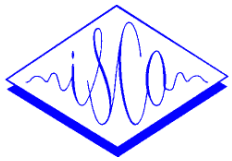

ISCA & IEEE Workshop on Spontaneous
Speech Processing and Recognition
Tokyo Institute of Technology, Tokyo, Japan
April 13-16, 2003

Key-word Spotting Using Phonetic Distinctive Features Extracted from Output of an LVCSR Engine

Tsuneo Nitta, Shingo Iseji, Takashi Fukuda, Hirobumi Yamada, and Kouichi Katsurada

Graduate School of Eng., Toyohashi University of Technology
1-1 Hibariga-oka, Tempaku, Toyohashi 441-8580, JAPAN
nitta@tutkie.tut.ac.jp

ABSTRACT

In this paper, we attempt to adopt a general-purpose LVCSR engine designed for dictation as a spoken dialogue recognition system. In the proposed system, a phoneme string output from the LVCSR engine is converted into a sequence of vectors represented with distinctive features (DF), then keywords assigned by a dialogue manager are detected from the input vector sequence using dynamic time warping (DTW). The proposed system takes advantage of the potential abilities of: (1) precise phoneme discrimination achieved by relaxing the linguistic constraint in the LVCSR engine, and (2) coping with the issues of substitution, deletion and insertion errors by combining a process of conversion from a phoneme into a distinctive feature vector and a key-word spotting process. The proposed system is compared with the general-purpose LVCSR engine in an experiment with a spoken dialogue corpus of a map guidance task and shows significant improvements. Comparative studies on language models and acoustic scoring procedure in key-word detection are also discussed with sub-word model and with confusion matrix, respectively.

1. INTRODUCTION

In the area of automatic speech recognition (ASR), dictation software packages are already available, but the packages need a well-matched language model for a topic, which makes it difficult to apply such software to tasks where the topic changes, such as web page access. In addition, spoken dialogue recognition encounters many problems such as breathing, rephrasing, hesitation, and poor grammar structure. Although key-word spotting is a powerful approach to overcome these issues [1], [2], it introduces new problems of frequent insertion errors and the increase of computation, and so can only be applied to small-vocabulary word recognition tasks in practice.

To address these problems, firstly we examine the use of a large-vocabulary continuous speech recognition (LVCSR) engine. LVCSR engines have capability for recognizing phonemes in any utterance, but the performance is lowered by linguistic constraints. **Figure 1** shows examples of the output phoneme strings with different types of language models (LM) in a Japanese LVCSR engine with a vocabulary of approximately 20,000 words. 3-gram outputs a grammatical sentence, however, the phoneme string is incorrect because of unknown words "Kentucky" and "fried". On the other hand, 0-gram that has no linguistic constraint produces a comparatively correct phoneme string, though the sentence is ungrammatical. In this paper, various types of n-gram, n=0, 1, 2, 3 are compared together with Japanese sub-word models. Secondly, to cope with the substitution, deletion and insertion errors in spontaneously spoken dialogue, we introduce a key-word spotting process and a preceding conversion process that converts a phoneme string into a sequence of distinctive feature vectors.

This paper is organized as follows. Section 2 outlines the implementation of a spoken dialogue recognition system, Section 3 describes the experimental setup and results, and provides a discussion, and Section 4 finishes with some conclusions.

2. SYSTEM OVERVIEW

Figure 2 shows a block diagram of the proposed system for spoken dialogue recognition. The system is divided into three processors: the front-end processor, which is implemented with a Japanese LVCSR engine "Julius" [3], the spoken language processor (SLP), and the dialogue manager (DM). In the proposed system, the only application-specific part is the dialogue scenarios in DM.

In the front-end processor, an input speech is sampled at 16 kHz and a 512-point FFT of the 25 ms Hamming-windowed speech segments is applied every 10 ms. The resultant FFT power spectrum is then integrated into 24-ch BPFs output with mel-

---

3 – gram: / geNzai, daizu jiki ni kitai suru /

2 – gram: / geNzai, ga iru jiki ni zeN suru /

1 – gram: / geNzai kitai to jishiN, saisen /

0 – gram: / keN ta ki fu ra izu chi kiN iki tai zu N /

<Input utterance>

/kentaqki: furaidochikiN ni ikitaiN desu kedomo/

(Where can I go to Kentucky Fried Chicken?)

Figure 1. Comparison of Phoneme Strings Output by Different LMs of a Japanese LVCSR engine

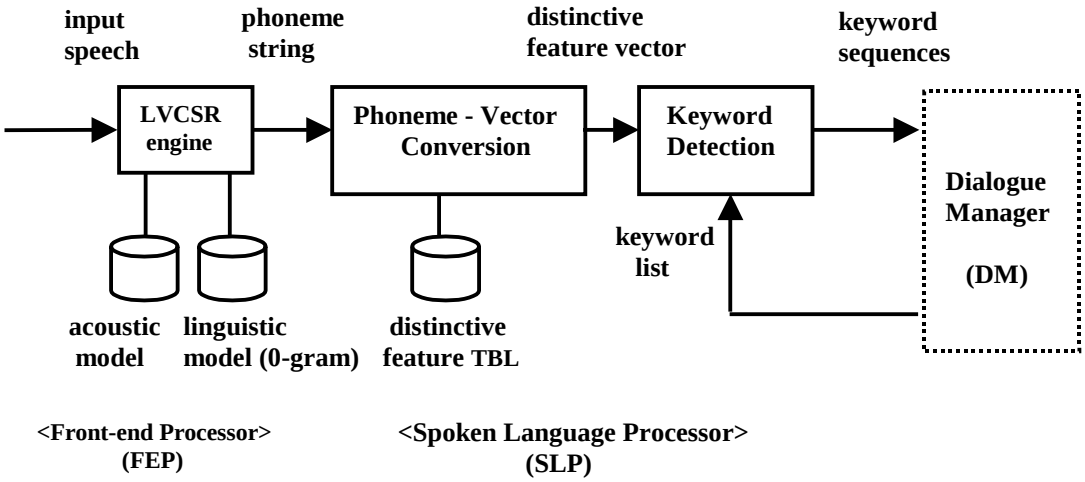

Figure 2 Block diagram of a spontaneous speech recognition system

-scaled center frequencies. Then, 25 feature parameters including 12 static parameters (mel-cepstrum), 12 dynamic features (Δ_t) and ΔP (logarithmic power) are extracted after converting the output of BPFs into cepstrum coefficients (MFCC) by using DCT. MFCC parameters are processed with CMN for every utterance. An HMM classifier has a tri-phone model with 2,000 states in total, and each output probability is represented in the form of Gaussian mixtures (mix=16), and diagonal matrices are used. The original LVCSR engine "Julius" performs a two-pass search using a 2-gram model and a 3-gram model on the respective passes, however, we eliminate the linguistic constraint and use a 0-gram model in both passes.

The SLP detects keywords, which are given by the DM, in the 1-best phoneme string output from the front-end processor. Firstly the phoneme string is converted into a sequence of vectors represented with distinctive features. We introduce 12 Japanese distinctive features (high, back, low, anterior, coronal, obstruent, voiced, continuant, nasal; two articulatory features of "vocalic/ non-vocalic" and "consonantal/ non-consonantal" were replaced by "semi-vowel (/j, w, r/)/ non-semi-vowel" and "fricative (/s, z, h/)/non-fricative", then added "vocalic/ consonantal").

Secondly the distinctive feature vector at frame i x(m, i), m=1,2,…,12, i=1,2,…,I is matched with keywords assigned by the DM after converting phoneme k in a keyword into a distinctive feature vector at frame j $r_k(m, j)$ k=1,2,…,29, j=1,2,…,J. We use the following equation as the distance between vectors:

$$d_k(i, j) = \sum_{m=1}^{12} \{ x(m, i) - r_k(m, j) \}^2 \qquad (1)$$

$$g(i, j) = \min \begin{cases} g(i-1, j) + d_k(i, j) & (a) \\ g(i-1, j-1) + 2d_k(i, j) & (b) \\ g(i, j-1) + d_k(i, j) & (c) \end{cases} \qquad (2)$$

$$c(i, j) = \begin{cases} c(i-1, j) + 1 & \text{if (a)} \\ c(i-1, j-1) + 2 & \text{if (b)} \\ c(i, j-1) + 1 & \text{if (c)} \end{cases} \qquad (3)$$

$$D(i) = g(i, J)/c(i, J) \qquad (4)$$

where, $d_k(i, j)$ is Hamming distance, g(i, j) is accumulated distance, c(i, j) is the pass weight of DTW, and D(i) is the distance of the keyword. $d_k(i, j)$ between vowels is doubled. After executing end-point free DTW, keywords with D(i) under the threshold are detected. In addition, words overlapped with each other are suppressed to eliminate redundant insertions.

There are other approaches such as confusion matrix (CM) [4] to match the phoneme string with reference strings, however, because CM depends on acoustic environment we use the approach described above. The distinctive features explicitly characterize the property in speech production and can express those phonemes for which the manner and/or place of articulation is similar, as close distance vectors.

The DM responds to a user along the dialogue scenario corresponding with the accepted key-word sequences. In the following section, experiments of keyword recognition with a spoken dialogue corpus and the results are discussed.

3. EXPERIMENTS

3.1 Spoken Dialogue Corpus

The following corpus was used for the evaluation test:

A subset of a spontaneously spoken dialogue corpus "ETL Map Guidance Task" collected using a WOZ system [5], consisting of 100 utterances spoken by 14 unknown male and female speakers.

3.2 Experimental Setup

Keyword detection performance was evaluated with various types of language models (LM) of the LVCSR engine "Julius". **Table 1** shows the detailed parameters of LM and scoring. In the table, the score of a sentence h composed of n words $\{w_1,w_2,\ldots,w_n\}$ is calculated by the following equation:

$$S(h) = L_{AM} + W\, L_{LM} + n*Ip \qquad (5)$$

where, L_{AM} and L_{LM} are the logarithmic output-probability of the acoustic model (AM) and the logarithmic occurrence-probability of the LM corresponding to the sentence, respectively. W, n*, and Ip are the weight of LM, number of words in a sentence, and word insertion penalty, respectively.

The size of vocabulary is 20,000 words. 109 keywords were selected from the test corpus, of which Julius contains 66 words (enrolled) and the other 43 are unknown words. An evaluation test is also done by enrolling all the keywords into the lexicon of Julius. In the test corpus, there exist some unknown, non-keywords which often occur at the beginning and end of a spoken sentence.

Evaluations using sub-word LMs [6] and CM (confusion matrix) were also conducted and the results were compared with the 20k word LM and DF, respectively.

3.3 Experimental Results

[A] Comparison of LM constraints

Table 2 shows the experimental results of keyword spotting. All the experiments were controlled to give approximately the same FA/WH (false acceptance per word and hour). In the table, "enrolled" means that 43 unknown keywords are enrolled into the lexicon and "n-enrolled" is n-gram after the enrollment of keywords. The results show that the 0-gram model, which is that with no LM constraint, gives the best recognition performance. The improvement effect of relaxing the LM constraint appears to be greater for unknown words than for enrolled words. The reason for the poor results of the 3-, 2-, and 1-gram LMs is because the input phoneme string is forced to run along the rigid rail of the n-gram LM, even if the AM can output more accurate phoneme strings. Here, the word error rate of the original LVCSR without SPL after enrolling all the keywords was 44.9%.

Figure 3 shows a comparison between before-enrollment and after-enrollment of keywords. We can see that the enrollment of keywords does not improve keyword recognition performance when we use the 0-gram LM, that is we can make the LVCSR application-independent.

[B] Comparison between 20k LM and sub-word LMs

Table 3 shows the experimental results of the comparison of 0-gram LMs. Japanese sub-word LMs of 1-syllable and 2-syllables are tested. In the table, "2-syllables (selected)" means only 2-syllable sub-words in the 20k-lexicon were picked up and merged with the 1-syllable sub-word LM. In the sub-word LMs, the 1-syllable sub-word LM and 2-syllable (selected) sub-word LM give the same performance.

Table 1. Linguisitic Constraints in LVCSR "Julius"

	1st Pass	2nd Pass	Weight (LM)	Insertion Penalty
3-gram:	2-gram	3-gram	8	-2
2-gram:	2-gram	2-gram	8	-2
1-gram:	1-gram	1-gram	8	-2
0-gram:	0-gram	0-gram	0	-5

Table 2. Keyword Recognition Result: Various LM Constraints

	substitution	deletion	word error rate [%]			FA/WH
				detail		
				enrolled	unknown	
3 - gram	**29**	**19**	**18.9**	**16.3**	**26.6**	**46.9**
2 - gram	**51**	**19**	**28.0**	**27.9**	**28.1**	**45.4**
1 - gram	**46**	**18**	**25.2**	**21.6**	**35.9**	**43.8**
0 - gram	**18**	**9**	**10.6**	**12.1**	**6.2**	**46.7**
3-enrolled	**25**	**11**	**14.2**	—	—	**45.7**
2-enrolled	**52**	**21**	**29.1**	—	—	**44.5**
1-enrolled	**43**	**16**	**23.2**	—	—	**43.9**
0-enrolled	**19**	**9**	**11.0**	—	—	**46.9**

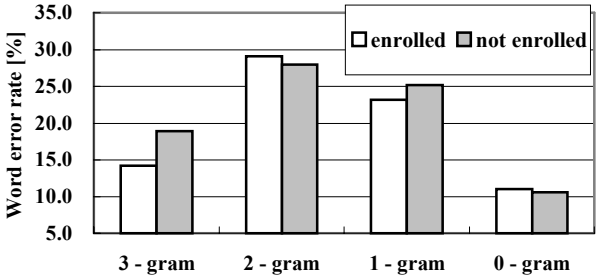

Figure 3. Comparison of LM Constraints and before/ after Keyword Enrollment

Table 3. Keyword Recognition Result: 20k LM vs. Sub-word LM

	substitution	deletion	WER [%]	FA/WH
20k-LM	**18**	**9**	**10.6**	**46.7**
1 syllable	**20**	**15**	**13.8**	**45.2**
2 syllables	**27**	**14**	**16.5**	**44.4**
2 syllables (selected)	**22**	**13**	**13.8**	**45.3**

Figure 4 shows the comparison between the 20k LM and the sub-word LMs. The results show the 20k LM outperforms the sub-word LMs. One of the reasons for this is shown in **Figure 5**. In this case, a user speaks "Then, how can I go to LOTTERIA?".

In the figure, the 20k LM has sub-words "Rotte" and "Lia (Lear)" of an unknown word "LOTTERIA". Thus it seems the 20k LM sometimes contributes to output correct words or the sub-word of a word.

[C] Comparison between DF and CM (confusion matrix)

Figre 6 shows the comparison between DF and CM that are used for acoustic scoring in DTW. Here, the procedure, in which the distance $d_k(i, j)$ between vowels is doubled, is removed for comparison. CM was calculated with the subset of the same dataset (JNAS: Japanese Newspaper Article Sentences) used in acoustic model (AM) design of the LVCSR engine. The results show DF outperforms CM under the different acoustic environment.

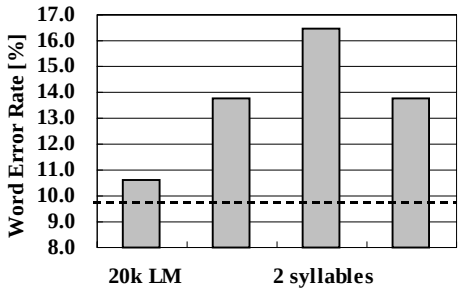

Figure 4. 20k LM vs. Sub-word LMs

4. CONCLUSION

In this paper, we proposed a novel recognition algorithm for spotting keywords in a spontaneously spoken dialogue. The proposed algorithm:

(a) derives the potential abilities of phoneme discrimination in LVCSR by relaxing the LM constraint, and

(b) can accurately detect keywords in spoken dialogue using DTW between distinctive feature vectors with less computation time.

We also pointed out that the 20k LM of a general-purpose LVCSR outperforms the Japanese sub-word LMs and that DF gives more robust performance than CM. In future work, we will implement the proposed algorithm into a multi-modal interaction system [7] and investigate the performance in practical environments.

20k LM:

/ ja aq daq roqte ria ya do eq cha ri N N desho ka /

1-syllable:

/ ja a zu do u te ri a e e a a do i ja i N de sho u ka /

2-syllables:

/ nida a do pu ege ya era a do ita i N de sho u ka /

2-syllables (selected):

/ ja a do q te ria ya a doi q zai Nde shou ka /

< Input utterance >

/ jaa: roqteria ewa dou iqtara iiN de shouka /

(Then, how can I go to LOTTERIA ?)

Figure 5. Comparison of Phoneme Strings Output by 20k LM and Sub-word LMs

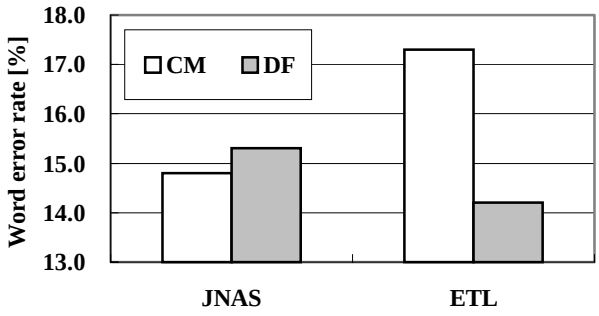

Figure 6. Confusion Matrix vs. Distinctive Feature

REFERENCES

[1] J. R. Rohlicek, W. Russel, S. Roucus, and H. Gish, "Continuous HMM for Speaker Independent Word Spotting," Proc. ICASSP, pp.627-630 (1994.5).

[2] H. Matsu'ura, Y. Masai, J. Iwasaki, S. Tanaka, H. Kamio, and T. Nitta, "A Multi-modal, Keyword-based Spoken Dialogue System – MultiksDial," Proc. ICASSP, pp.II-33-36 (1994.4).

[3] A. Lee, T. Kawahara, and K. Shikano, "Julius – an Open Source Real-Time Large Vocabulary Recognition Engine," Eurospeech2001, pp.1691-1694 (2001).

[4] S. Makino, S. Homma, and K. Kido, "Speaker Independent Word Recognition System Based on Phoneme Recognition for a Large Size (212 words) Vocabulary," J. Acoust. Soc. Jpn., (E) 6, 3, pp.171-180 (1985).

[5] K. Ito, T. Akiba, S. Hayamizu, and K. Tanaka, "A Spontaneous Speech Dialogue Corpus Collected Using WOZ System," Proc. Acoustic Society of Japan – Autumn Meeting, 1-1-19, pp.37-38 (1998.9) (in Japanese).

[6] K. Ng, "Toward Robust Methods for Spoken Document Retrieval," Proc. ICSLP, pp.939-942 (1998.11).

[7] K. Katsurada, Y. Ootani, Y. Nakamura, S. Kobayashi, H. Yamada, and T. Nitta, "A Modality-Independent MMI System Architecture," ICSLP, pp.2549-2552 (2002.9).