RECENT TOPICS IN SPEECH RECOGNITION RESEARCH AT NTT LABORATORIES

Sadaoki Furui, Kiyohiro Shikano, Shoichi Matsunaga, Tatsuo Matsuoka, Satoshi Takahashi, and Tomokazu Yamada

NTT Human Interface Laboratories
3-9-11, Midori-cho, Musashino-shi, Tokyo, 180 Japan

ABSTRACT

This paper introduces three recent topics in speech recognition research at NTT (Nippon Telegraph and Telephone) Human Interface Laboratories.

The first topic is a new HMM (hidden Markov model) technique that uses VQ-code bigrams to constrain the output probability distribution of the model according to the VQ-codes of previous frames. The output probability distribution changes depending on the previous frames even in the same state, so this method reduces the overlap of feature distributions with different phonemes.

The second topic is approaches for adapting a syllable trigram model to a new task in Japanese continuous speech recognition. An approach which uses the most recent input phrases for adaptation is effective in reducing the perplexity and improving phrase recognition rates.

The third topic is stochastic language models for sequences of Japanese characters to be used in a Japanese dictation system with unlimited vocabulary. Japanese characters consist of Kanji (Chinese characters) and Kana (Japanese alphabets), and each Kanji has several readings depending on the context. Our dictation system uses character-trigram probabilities as a source model obtained from a text database consisting of both Kanji and Kana, and generates Kanji-and-Kana sequences directly from input speech.

1. PHONEME HMM CONSTRAINED BY STATISTICAL VQ-CODE TRANSITION

1.1 Introduction

Speaker-independent phoneme models need a large amount of training data to cover the phonetic features of various speakers and various phoneme environments. However, more training data leads to broader spectral feature distributions of each phoneme. One speaker's spectral feature distribution often overlaps the distributions of different phonemes of other speakers. This causes confusion and degrades recognition performance.

It has widely been confirmed that transitional spectral information, such as that represented by the so-called delta-cepstrum, is effective for decreasing these overlaps and improving the performance of speaker-independent recognition when it is used together with instantaneous spectral information [1]. The delta-cepstrum attempts to model the differential spectrum. The second-order differential spectrum [2][3] has also been used to further improve the performance.

In the vector quantization (VQ) -based recognition, another kind of transitional spectral information can be represented by VQ-code sequences. Conditional models of VQ-code transitions have been proposed to obtain accurate speech models [4][5]. However, it is very difficult to obtain conditional models from the training data in a real situation, since numerous parameters must be estimated. We have tried to use bigrams of VQ-code sequences to represent statistical transitional information and restrict the feature distributions to a suitable region [6]. This method reduces the overlap of feature distributions between phonemes without requiring a huge amount of training data.

1.2 Bigram-constrained HMM

A bigram-constrained HMM is obtained by combining a VQ-code bigram and the conventional HMM. The output probability distribution of the model changes depending on the VQ-code of the previous frame even in the same state. A block diagram of the procedure generating the bigram-constrained HMM is shown in Fig. 1.

First, a universal codebook is generated from a large amount of speech data consisting of utterances of many speakers, and conventional speaker-independent phoneme HMMs are trained using this codebook. Speech data for calculating a VQ-code bigram is collected and fuzzy-vector-quantized using the universal codebook. The VQ-code bigram probability is given by

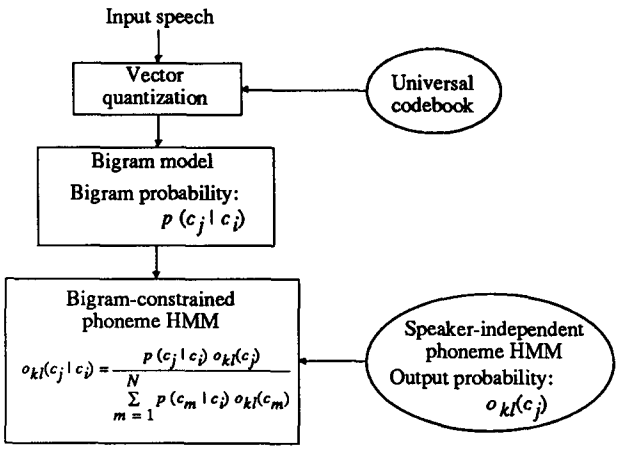

Fig. 1. Block diagram for generating bigram-constrained HMM

$$p(c_j \mid c_i) = \frac{\sum_t u(y_{t-1}, c_i)\, u(y_t, c_j)}{\sum_t \sum_m u(y_{t-1}, c_i)\, u(y_t, c_m)} \qquad (1)$$

where c_j and c_i are VQ-codes of the current and the preceding frames, respectively. Here, $u(y_t, c_j)$ is the membership value of the VQ-code c_j for feature vector y_t.

The output probability of each VQ-code associated with the transition from state k to state l is calculated as a conditional probability according to the preceding frame VQ-code, such as

$$o_{kl}(c_j \mid c_i) = \frac{p(c_j \mid c_i)\, o_{kl}(c_j)}{\sum_{m=1}^{N} p(c_m \mid c_i)\, o_{kl}(c_m)} \qquad (2)$$

where $o_{kl}(c_j)$ is the output probability of the current frame VQ-code c_j for the transition from state k to state l, and N is the codebook size.

There are several types of bigram-constrained HMMs depending on the method of calculating the VQ-code bigram. A speaker-dependent bigram-constrained HMM is obtained by using speech data of an input speaker for the bigram calculation. A speaker-independent bigram-constrained HMM, on the other hand, is obtained by using speech data of many speakers different from the input speaker. Moreover, the bigram can be calculated separately for each phoneme (phoneme-dependent bigram) or jointly for all phonemes (phoneme-independent bigram).

1.3 Experimental Results

The proposed method was evaluated by an 18-Japanese-consonant recognition task. The 5240-Japanese word sets uttered by 10 males and 10 females were used. Phoneme periods extracted from the even-numbered words by 16 speakers were used for training the conventional HMMs, and those from odd-numbered words of the other four speakers were used for evaluation. 216 phonetically-balanced-Japanese-word sets uttered by the four test speakers were used to calculate speaker-dependent bigrams. A speaker-independent bigram was obtained using all the training utterances by the 16 training speakers.

Multiple codebooks were created for each set of the feature parameters: 16 cepstrum coefficients, 16 delta cepstrum coefficients, and delta energy. The frame period for feature extraction was 8 ms. Codebook sizes were 256, 256, and 64, respectively. The VQ-code bigrams were calculated independently for each codebook. Phoneme-dependent bigrams were calculated referring to manually segmented phoneme labels. The HMMs had four states and three loops. Each phoneme had two models, one for the beginning and the other for the middle of words.

Average phoneme recognition rates for various bigram conditions are shown in Table 1. It can be concluded that the phoneme-dependent bigram is much better than the phoneme-independent bigram. The recognition rate using the phoneme- and speaker-dependent bigrams achieved 78.6%, which is 7.8% higher than that obtained by the traditional HMM without combining the bigrams. Even the speaker-independent bigram can improve the recognition rate by 5.5%.

Table 1 - Phoneme recognition rate

		Phoneme-independent	Phoneme-dependent
Bigram-constrained HMM	Speaker-independent	73.8%	76.3%
	Speaker-dependent	74.9%	78.6%
Conventional HMM (speaker-independent)		70.8%	

These experiments confirm the effectiveness of the bigram-constrained HMM, with which output probabilities are conditioned by the VQ-code bigram.

2. TASK ADAPTATION IN STOCHASTIC LANGUAGE MODELS FOR CONTINUOUS SPEECH RECOGNITION

2.1 Introduction

One of the ultimate goals of automatic speech recognition is to create a device capable of transcribing speech into written text. The most typical structure of the recognizer consists of an acoustic processor and a linguistic decoder. Most of the recent linguistic decoders use stochastic language models, such as bigrams and trigrams of linguistic units. In order to obtain a reliable stochastic language model, which achieves good recognition performance, it is necessary to use a very large text database. It is also necessary that the task of the database is similar to the recognition task. When the recognition task is changed, recognition performance decreases because the language model is no longer appropriate. However, it is not always possible to obtain a very large text database for each new task. Therefore, it is very important to establish a method of adapting the statistical language model to a new task using a small amount of text similar to the recognition task.

2.2 Model Adaptation

We have investigated two approaches for adapting a syllable-trigram model to a new task in a Japanese transcription system, a phonetic typewriter, based on continuous speech recognition [7]. In this system, sentences are assumed to be spoken phrase by phrase. Japanese syllables, which are basic linguistic units, roughly correspond to consonant-vowel concatenation units. The first adaptation method, "preliminary learning", uses a small amount of text similar to the recognition task, and the second method, "successive learning", is based on supervised learning using the most recent input phrases. Since the goal of the system is to transcribe speech into written text, recognition errors are finally corrected by the user. Therefore, supervised learning can be applied using text which has recently been input to the system.

The successive learning method using "cache" text was first proposed by Kuhn et al. for a stochastic language model based on a word-trigram model [8]. They showed that this method greatly reduced the test-set perplexity. We applied this method to the syllable-trigram models.

An initial syllable-trigram model based on a large text database on a specific task or on a general task covering several fields is assumed to be given. Figure 2 shows the adaptation approaches for trigram models by preliminary learning and successive learning. On the right-hand side of the figure, the top row corresponds to successive learning and the second row corresponds to preliminary learning. The adapted trigram is generated using the deleted interpolation technique.

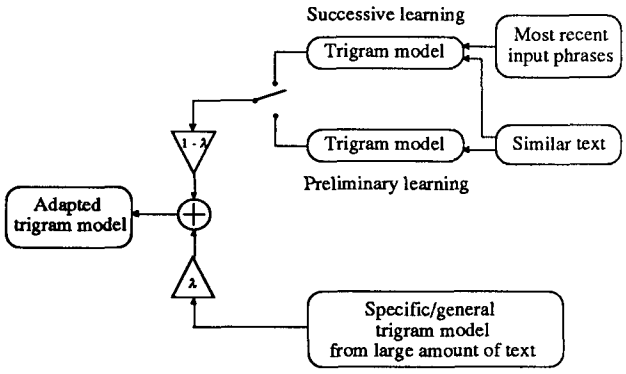

Fig. 2. Adaptation of trigram models

2.3 Experimental Results

The effect of each adaptation method was evaluated with syllable perplexities and phrase recognition rates. Two large text databases about conference registration (1.4 × 104 kbytes, 9.3×10^4 phrases) and about travel arrangement (1.1×10^4 kbytes, 7.9×10^4 phrases) were used in the experiments. The recognition task concerned conference registration. The travel arrangement database was used to generate an initial trigram model on a specific task different from the recognition task.

In successive learning, the initial trigram model generated from the travel arrangement database was modified using the most recent 100 phrases at every fixed number of input phrases. Since the number of available phrases for the first 100 input phrases was less than 100, phrases of the similar task were added to keep the total number of training phrases at 100, as shown in Fig. 3.

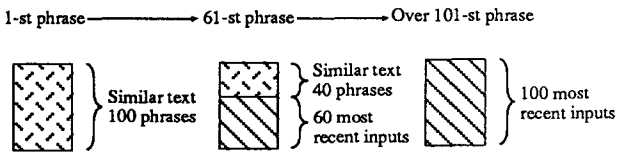

Fig. 3. Construction of learning text in successive learning

The recognition process flow of the phonetic typewriter is as follows: Cepstra, delta-cepstra and delta-energy are extracted for each frame of input speech and are fuzzy-vector-quantized. Phoneme sequence likelihood is then calculated as a joint likelihood combining acoustic and syntactic likelihoods. The acoustic likelihood is derived from phoneme-based HMMs, and the syntactic one is obtained by a predictive LR parser [9] and the syllable trigram. Each HMM is trained by word utterances. The joint likelihood is maximized to obtain the solution.

As a reference, speaker-dependent recognition tests were first carried out on 279 phrases uttered by one male speaker. The trigram model was generated from the large conference registration text database, which is the same task as the recognition task. The syllable perplexity and the phrase recognition rate were 12.2 and 64.2%, respectively. These values were the targets for the adaptation.

Table 2 shows syllable perplexities and phrase recognition rates for various learning conditions. For the successive learning case, the perplexities are shown as a function of the learning period. The perplexity was reduced from 24.5 to 18.1 by the adaptation using 100 phrases of the similar text, and was reduced to 14.6 by successive learning at every 10 phrases using the most recent 100 phrases. This clearly shows that successive learning is more effective than preliminary learning, and that the more frequent the successive learning is, the more effective it becomes.

Table 2 - Syllable perplexity and phrase recognition rate

Learning method		Perplexity	Recognition rate
No adaptation		24.5	42.3%
Preliminary learning		18.1	46.6%
Successive learning	every 30 phrases	15.8	-
	every 20 phrases	15.4	-
	every 10 phrases	14.6	50.9%
	every 5 phrases	14.4	-

A recognition experiment for successive learning was conducted with learning at every 10 phrases. The recognition rates were improved from 42.3% to 46.6% by preliminary learning and to 50.9% by successive learning. Although still there is a gap between the performances based on training using a large text database and adaptation, these results confirm that the successive learning method is effective.

3. CHARACTER SOURCE MODELING FOR A JAPANESE DICTATION SYSTEM

3.1 Introduction

Japanese sentences are usually written using both Kana (Japanese alphabets) and Kanji (Chinese characters). Kana are the minimal linguistic units in the written form and correspond to Japanese syllables, which consist of a consonant-vowel pair or a single vowel. Kanji are linguistic units having one or more meanings and pronunciations, and the pronunciations can be written by Kana sequences. Japanese words are made up of sequences of Kana and Kanji. For convenience we will use "Kanji" to represent both Kana and Kanji.

In English, word sequence probability is usually used to make a language model. However in Japanese, since words are not clearly defined, Kana sequence probability has usually been effectively used for speech recognition. We are trying to build a Japanese dictation system using a "Kanji" source model, instead of using a Kana source model, for the following reasons [10][11].

1) For a given length of character source, a "Kanji" source model can effectively deal with a longer phoneme context.
2) A "Kanji" source model can directly convert speech into Kana and Kanji sequences, without post-processing of Kana-to-Kanji conversion.

3.2 Character Source Modeling

A "Kanji"-trigram probability is calculated using a text database to construct a character source model. Since ordinary Japanese texts use several thousand different "Kanji", the trigrams obtained using practical databases are very sparse. To alleviate this problem, the deleted interpolation algorithm is used. That is, the improved trigram $\hat{P}^{(3)}$ is estimated by linear combination of a zerogram $P^{(0)}$, unigram $P^{(1)}$, bigram $P^{(2)}$, and trigram $P^{(3)}$:

$$\hat{P}^{(3)} \equiv \lambda_0 P^{(0)} + \lambda_1 P^{(1)} + \lambda_2 P^{(2)} + \lambda_3 P^{(3)} \qquad (3)$$

Test-set perplexities and the number of different characters for three different tasks are listed in Table 3. The task of the recognition test data is the conference registration. When the tasks of training and test data are the same, the Kana-based perplexities of "Kanji" source models are smaller than those of Kana source models. The results

shown in the table indicate that a "Kanji" source model is efficient for the Japanese dictation system, and that the source model is highly dependent on the task.

Table 3 - Test-set Kana-based perplexity for text database and number of different characters

Text database for training	Kana-based perplexity		Number of different characters	
	Kana	"Kanji"	Kana	"Kanji"
Conference registration	10.5	9.7	117	1362
Travel arrangement	18.6	31.3	114	1480
Both	9.6	10.1	120	1696

3.3 Japanese Dictation System

Figure 4 is a schematic diagram of the dictation system. This system dictates phrase-by-phrase input speech using the HMM-LR method. HMMs are used for phoneme recognition, and a "Kanji" source model and a predictive LR parser are used for the language processing. The predictive LR parser predicts a phoneme of the input speech successively from left to right (from the beginning to the end) according to the context-free rewriting rules, and sends it to the HMM phoneme verifier. The phoneme verifier calculates the likelihood of the predicted phoneme for the input speech, and returns the score to the LR parser. In the reduce action of the LR parser, a phoneme sequence is converted into a "Kanji", based on the weighted sum of the HMM likelihood and the trigram likelihood.

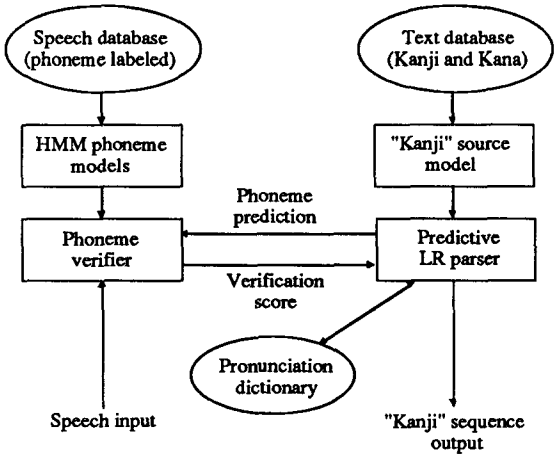

Fig. 4. Schematic diagram of Japanese dictation system

Each Kanji character has several readings depending on the context. The "Kanji" trigram, however, is calculated from only the character sequences in the training text database, neglecting the reading of the "Kanji", and context-independent rewriting rules for a "Kanji"-to-phoneme sequence are given to make an LR table. Therefore, the parser produces many contextually wrong candidates. To solve this problem, we added the step of consulting a dictionary to check the phoneme sequence of the candidate and eliminated the candidates whose phoneme sequences were inappropriate to the "Kanji" sequence. The test-set Kana-based perplexities for the "Kanji" source models with and without a pronunciation check using a dictionary are listed in Table 4.

Table 4 - Test-set Kana-based perplexity for "Kanji" source models

Text database for training	Kana-based perplexity	
	Without dictionary	With dictionary
Conference registration	9.7	7.7
Travel arrangement	31.3	25.7
Both	10.1	8.0

3.4 Experimental Results

Speaker-dependent transcription experiments were performed. HMM phoneme models were made from 5240 Japanese words and 216 phonetically balanced words spoken by a male speaker. The "Kanji" source model was obtained from the text database of the conference registration task. Test data consisted of 274 phrases uttered by the same speaker.

The transcription rates (top and top four) are shown in Table 5. A correct phrase, here, means an output phrase candidate whose "Kanji" sequence and pronunciation are both correct, and the character transcription rate is calculated by the summation of correct output characters, neglecting insertion and deletion. These results indicate that the proposed method of pruning based on the "Kanji" sequence pronunciation is effective in eliminating candidates whose readings do not fit the context.

We are also trying another method using a pronunciation-tagged "Kanji" source model to further reduce erroneous outputs that have inappropriate readings of "Kanji" [11].

Table 5 - Phrase and character transcription rate

Dictionary	Transcription rate		
	Phrases		"Kanji" characters
	(1st)	(1st - 4th)	
Without	58.4%	70.8%	71.2%
With	63.9%	74.5%	78.5%

4. DISCUSSION

Three recent topics in speech recognition research at NTT Human Interface Laboratories were introduced in this paper. We are still continuing our investigations of these topics to improve the recognition performances. Other topics in progress, but not mentioned here, include research on spontaneous speech recognition, neural-network-based approaches, HMM training techniques, new evaluation methods of continuous speech recognition, and speaker recognition.

REFERENCES

[1] S. Furui: "Speaker-independent isolated word recognition using dynamic features of speech spectrum", IEEE Trans. ASSP-34, 1, pp.52-59 (1986)

[2] S. Furui: "Cepstral analysis technique for automatic speaker verification", IEEE Trans. ASSP-29, 2, pp.254-272 (1981)

[3] H. Ney: "Experiments on mixture-density phoneme-modelling for the speaker-independent 1000-word speech recognition DARPA task", Proc. IEEE ICASSP 90, S13.9, pp.713-716 (1990)

[4] P. F. Brown: "The acoustic-modeling problem in automatic speech recognition", Doctoral thesis, CMU (1987)

[5] C. J. Wellekens: "Explicit correlation in hidden Markov model for speech recognition", Proc. IEEE ICASSP 87, 10.7, pp.384-386 (1987)

[6] S. Takahashi, T. Matsuoka and K. Shikano: "Phonemic HMM constrained by statistical VQ-code transition", Proc. IEEE ICASSP 92 (1992) (to be published)

[7] S. Matsunaga, T. Yamada and K. Shikano: "Language model adaptation for continuous speech recognition", 1991 IEEE-SPS Arden House Workshop on Speech Recognition, 8.2 (1991)

[8] R. Kuhn and R. DeMori: "A cache-based natural language model for speech recognition", IEEE Trans. PAMI-12, 6, pp.570-583 (1990)

[9] T. Hanazawa, K. Kita, S. Nakamura, T. Kawabata and K. Shikano: "ATR HMM-LR continuous speech recognition system", Proc. IEEE ICASSP 90, S2.4, pp.53-56 (1990)

[10] T. Yamada, T. Hanazawa, T. Kawabata, S. Matsunaga and K. Shikano: "Phonetic typewriter based on phoneme source modeling", Proc. IEEE ICASSP 91, S3.4, pp.169-172 (1991)

[11] T. Yamada, S. Matsunaga and K. Shikano: "Japanese dictation system using character source modeling", Proc. IEEE ICASSP 92 (1992) (to be published)