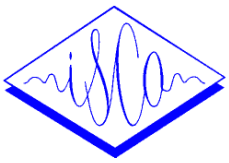

2nd International Conference on
Spoken Language Processing (ICSLP 92)
Banff, Alberta, Canada
October 12-16, 1992

A SPEAKER ADAPTATION BASED ON CORRECTIVE TRAINING AND LEARNING VECTOR QUANTIZATION

Myoung-Wan KOO and Chong-Kwan UN *

Basic Research Section 1
Korea Telecom Research Center
17 Umyon-dong, Seocho-gu, Seoul, 137-140, Korea

* Department of Electrical Engineering
Korea Advanced Institute of Science and Technology
373-1 Kooseong-dong, Yuseong-gu, Daejeon, 305-701, Korea

ABSTRACT

In this paper, we present a speaker adaptation technique based on corrective training(CT) and learning vector quantization(LVQ). Our algorithm consists of two stages: codebook adaptation and hidden Markov model(HMM) parameter adaptation. In the stage of codebook adaptation, we propose a codebook adaptation scheme using a neurally-inspired LVQ with highly discriminant ability. In the stage of HMM parameter adaptation, we propose a modified corrective training algorithm for speaker adaptation in which the HMM parameter adaptation obtained by probability transformation matrix are re-estimated to maximize the recognition rate on the adaptation speech. With this method, the recognition rate for new speakers can be improved.

10.21437/ICSLP.1992-183

I. INTRODUCTION

For speech recognition, the HMM typically requires a large amount of computation for training the system to obtain reliable probability estimates. The purpose of speaker adaptation is to obtain an acceptable recognition rate even for speakers who have not provided enough speech to train the recognition system. One method of speaker adaptation in a speech recognition system based on the HMM is the spectral mapping algorithm. This algorithm consists of two stages: codebook adaptation and HMM parameter adaptation [1]. In the stage of codebook adaptation, a codebook is generated for a new speaker by clustering a small size of target speech (adaptation speech). In the stage of HMM parameter adaptation, well-trained HMMs are transformed from a reference speaker to a new speaker using a probability transformation matrix.

The spectral mapping technique so far used has some drawbacks. One is that the codebook for a new speaker is made from adaptation speech. which is relatively shorter than training speech. Therefore, it is desirable to compensate for the insufficient amount of speech from the new speaker. Another one is that the HMM parameters for a new speaker are obtained by linear transformation from those of the reference speaker. In practice, however, the HMM parameters for a new speaker are not merely modeled in this way.

In this paper, we present a speaker adaptation technique based on CT and LVQ. By the proposed scheme, the codebook is generated to have discriminant feature rather than mininum distortion for adaptation speech and to compensate for the short size of adaptation speech. And the HMM parameters obtained by a probability transformation matrix are re-estimated to maximize the recognition rate on the adaptation speech.

II. CODEBOOK ALGORITHM BASED ON LVQ

2.1. LVQ

There are two kinds of LVQ's, LVQ1 and LVQ2. These have been used in pattern classification problems [2]. LVQ1 is the first version of LVQ, and LVQ2 isthe upgraded version of LVQ1. In LVQ1, for a given adaptation speech vector $\mathbf{x}$, codebook vector $\mathbf{m}_c$ is updated as

$$\mathbf{m}_c(t+1) = \mathbf{m}_c(t) + \alpha(t)(\mathbf{x}(t) - \mathbf{m}_c(t)) \qquad (1)$$

if $\mathbf{x}$ and the closest codebook vector belong to the same class;

$$\mathbf{m}_c(t+1) = \mathbf{m}_c(t) - \alpha(t)(\mathbf{x}(t) - \mathbf{m}_c(t)) \qquad (2)$$

if $\mathbf{x}$ and the closest codebook vector belong to different classes, where t is a discrete-time index (integer), and α(t) ($0 < \alpha(t) < 1$) decreases monotonically with time. The other codebook vectors are not modified. In LVQ2, if codebook vectors $\mathbf{m}_i$ and $\mathbf{m}_j$ are the nearest and next-to-nearest ones to $\mathbf{x}$, respectively, and $\mathbf{x}$ belongs to class C_j (but not class C_i) and also falls into window $\mathbf{w}$, then they are corrected as

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) - \alpha(t)(\mathbf{x}(t) - \mathbf{m}_i(t)) , \qquad (3)$$

$$\mathbf{m}_j(t+1) = \mathbf{m}_j(t) + \alpha(t)(\mathbf{x}(t) - \mathbf{m}_j(t)) .$$

We take α(t) as $\alpha(t) = \alpha_0 \times (1.0 - t/NMAX/24.0)$, where α_0 is emphirically determined to be 0.01 for LVQ1, 0.1 for LVQ2 and $NMAX$ is the maximum frame number, and window $\mathbf{w}$ is taken as $\mathbf{w} = | \mathbf{m}_i(t) - \mathbf{m}_j(t) |$.

2.2. Codebook adaptation based on LVQ

The adaptation speech should be labeled for the learning of LVQ since LVQ is a supervised learning algorithm. Here, we use

DTW between the adaptation speech and the reference speech in order to label the adaptation speech. Since the reference speech is already labeled and saved, we can find the correspondence between unlabeled speech and labeled speech by DTW. We use the K-means algorithm to initialize the codebook vectors for LVQ. Two kinds of codebooks are used as the initial codebook for LVQ. The first codebook is *K-means all* codebook which is trained with adaptation speech by itself. This codebook is conventional K-means codebook. The second codebook is *K-means each* codebook which is produced by concatenating each codebook trained separately on each of phone classes. Since adaptation speech is already labeled by DTW, we can segment and bring together the adaptation speech phone by phone. Considering the occurrence frequency and the spectral characteristics of each phone, we assign the number of codewords to each phone.

III. HMM PARAMETER ADAPTATION BASED ON CT ALGORITHM

3.1. A modified corrective training for adaptation

CT algorithm is modified to be used in the adaptation procedure. The modified corrective algorithm consists of the estimation of initial HMM parameters and the re-estimation of them for which the procedures are similar to that of the conventional method [3]. The initial HMM parameters for a new speaker are estimated by performing the spectral mapping algorithm on the HMM parameters of the reference speaker and the initial expected number of counts C_s at state s is also found when the HMM parameters of the reference speaker are obtained from the training speech of the reference speaker as

$$p(k_i/s) = C_s(k_i)/C_s = C_s(k_i)/\sum_{l=1}^{l=N} C_s(k_l), \quad (4)$$

where $p(k_i/s)$ is the output probability that generates the output symbol k_i at state s of HMM, and $C_s(k_i)$ is the expected number of counts that generate the output symbol k_i at state s of HMM.

With these parameters and the initial expected number of counts, the HMM parameters are iteratively re-estimated with regard to adaptation speech until no corrections occur or a maximum number of iteration is reached. In order to re-estimate the model, the probability $P(u/x)$ of each adaptation utterance u with respect to each model x is first computed by the forward algorithm. If k is the true model of adaptation utterance u, each model x can be classified into three kinds of classes as

$$\begin{array}{ll} \text{error} & \text{if } P(u/k) - P(u/x) \le 0, \\ \text{near-miss} & \text{if } 0 < \log_{10} P(u/k) - \log_{10} P(u/x) \le \delta, \\ \text{correct} & \text{if } \log_{10} P(u/k) - \log_{10} P(u/x) > \delta, \end{array} \quad (5)$$

where the near-miss criterion δ is empirically set to be 30. The expected numbers of counts of adaptation utterance u with regard to a correct model k ($C_s^{uk}(\hat{k}_j)$) and an incorrect model x ($C_s^{ux}(\hat{k}_j)$), which generate the observation symbol $\hat{k}_j$ at state s of HMM, can also be simultaneously obtained for each pair (u, x) of adaptation utterance u and model x. We can also find the initial expected number of counts $C_s(\hat{k}_j)$ for a new speaker that generate the output symbol $\hat{k}_j$ at state s of HMM. These initial expected number of counts $C_s(\hat{k}_j)$ is obtained from the initial expected number of counts C_s and the given observation probabilities for a new speaker by using (4). The expected number of counts are updated as

$$C_s(\hat{k}_j) = C_s(\hat{k}_j) + r\beta_w C_s^{uk}(\hat{k}_j), \quad (6)$$

when $C_s^{uk}(\hat{k}_j)$ is obtained, and

$$C_s(\hat{k}_j) = C_s(\hat{k}_j) - r\beta_b C_s^{ux}(\hat{k}_j), \quad (7)$$

when $C_s^{ux}(\hat{k}_j)$ is obtained, where the scaling factor r is computed as

$$r = \begin{cases} 1 & \text{for error} \\ 1 - (\log_{10}(P(u/k)/P(u/x)))/\delta & \text{for near-miss} \\ 0 & \text{for correct} \end{cases} \quad (8)$$

and the within-class learning rate β_w is set to be 1 and the between-class learning rate β_b is varied. For each iteration of the modified CT algorithm, we re-estimate the observation probability for a new speaker as

$$p(\hat{k}_j/s) = C_s(\hat{k}_j)/\sum_{l=1}^{N} C_s(\hat{k}_j). \quad (9)$$

If $C_s(\hat{k}_j)$ in (9) is negative, $p(\hat{k}_j/s)$ is set to be 10^{-4}. Note that we know the correct models of adaptation speech because the spectral mapping algorithm is a supervised adaptation algorithm.

The reason why the expected number of counts C_s for the reference speaker can be used as the initial expected number of counts for a new speaker is that the expected number of counts is conserved in the HMM parameter adaptation procedure of the spectral mapping method. In practice, the probability $p(\hat{k}_j/s)$ that the new speaker produces a new quantized spectrum $\hat{k}_j$ at state s is obtained by the HMM parameter adaptation procedure as

$$\begin{aligned} p(\hat{k}_j/s) &= \sum_{i=1}^{N} p(k_i/s)p(\hat{k}_j/k_i) = \sum_{i=1}^{N} [C_s(k_i)/C_s]p(\hat{k}_j/k_i) \\ &= 1/C_s \sum_{i=1}^{N} C_s(k_i)P(\hat{k}_j/k_i) = C_s(\hat{k}_j)/C_s, \end{aligned} \quad (10)$$

where the probability $p(k_i/s)$ is the same as (4), and the probability for spectrum $\hat{k}_j$, given k_i, is assumed to be independent on s. The mapping probability $p(\hat{k}_j/k_i)$ for all i and j forms a probabilistic transformation matrix from one speaker's spectral space to another at each state. Comparing (10) with (4), we can know that the expected number of counts C_s for the reference speaker have direct relation with the expected number of counts $C_s(\hat{k}_j)$ for a new speaker that generates the output symbol $\hat{k}_j$ at state s.

3.2. HMM parameter adaptation based on CT algorithm

It is necessary to modify the speaker-dependent baseline system before the modified CT algorithm is performed on the adaptation speech. The baseline system should save the expected number of counts for the reference speaker in addition to HMM parameters. If a new speaker utters a small size of adaptation speech, the spectral mapping algorithm is first performed for obtaining the initial HMM parameters for the new speaker. This

algorithm consists of codebook adaptation and HMM parameter adaptation. The codebook for a new speaker is generated from adaptation speech itself as presented in Section 2.2.

The HMM parameters for a new speaker are obtained in two steps. In the first step, the initial HMM parameters are found in the process of HMM parameter adaptation of the spectral mapping algorithm. In order to find the initial HMM parameter for a new speaker, the probabilistic transformation matrix is obtained by counting the correspondence between the VQ codeword indices of reference speech and those of adaptation speech via the mapped codebook. Since the transformation between speakers cannot be modeled by a single transformation, we transform phoneme-dependently the speech of one speaker to that of another. In practice, we interpolate the phoneme-dependent transformation matrix $\mathbf{T}(ph)$ with the phoneme-independent transformation matrix $\mathbf{T}$ to get

$$\mathbf{T}_{ip} = \lambda(N_{ph})\,\mathbf{T}(ph) + [1 - \lambda(N_{ph})]\,\mathbf{T}, \qquad (11)$$

where $\mathbf{T}_{ip}$ indicates the interpolated transformation matrix, $\lambda(N_{ph})$ $= MIN(1, SQRT(N_{ph}/900))$ and N_{ph} is the number of observed frames of phoneme ph.

In the second step, we perform the modified CT algorithm on the initial HMM parameters obtained by the spectral mapping algorithm in order to refine those parameters.

IV. DESCRIPTION OF THE BASELINE SYSTEM

We first established a speaker-dependent baseline system using phoneme-like subword units (47 units) of Korean speech which consists of phonetically balanced 100 word vocabulary. Average number of syllables in the vocabulary was 1·6. Ten repetitions were pronounced by a male speaker who was designated as a reference speaker. The speech was sampled at 10 kHz and represented every 10 msec by three sets of parameters: (1) mel-scaled cepstral coefficients; (2) their differential coefficients; and (3) log power and differential log power. These parameters were vector quantized separately into three codebooks, each with 256 entries. The subword model of HMM has 7 states and 12 transitions with three output probability density functions as shown in Figure 1.

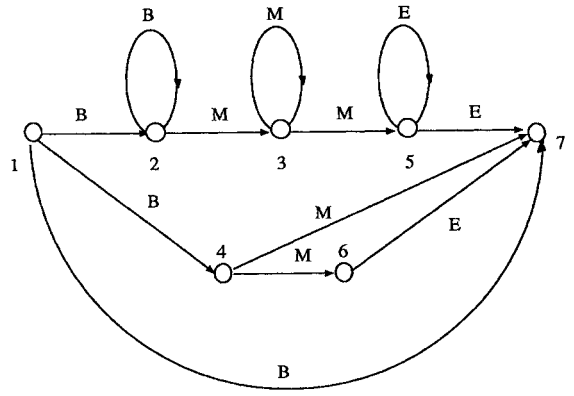

Fig. 1. HMM subword model

Three iterations of the forward-backward algorithms on the four repetitions of the manually segmented 100 word vocabularies were run and smoothed by the co-occurrence method [4] in order to initialize our phoneme-like units. We modeled each phone with a context-independent HMM after running another five iterations of the forward-backward algorithm on those vocabulary and another three repetitions of 100 words word by word. A recognition accuracy of 96.3% was obtained from the recognition test with other sets of 100 words.

V. EVALUATION EXPERIMENTS

Speaker adaptation experiments were conducted for two male and one female speakers excluding the reference speaker. Each speaker pronounced 100 words four times. We used one repetition of 100 words as adaptation speech and three repetitions of 100 words as test words.

First, we performed a speaker adaptation experiment based on LVQ. Since our system is based on three codebooks, we can find the LVQ codebooks for them, respectively. In this work, however, we consider only one LVQ codebook of which feature is cepstral coefficients. The LVQ codebook for a new speaker is generated in two ways. When the *K-means all* codebook is used to initialize the LVQ codebook, we compared the performance of the speaker adaptation approach with LVQ codebook to that of the scheme with the *K-means all* codebook, and also to that without adaptation. We also compared VQ distortion between the test speech and each codebook. The results are shown in Table 1.

TABLE 1
PERFORMANCE COMPARISON OF SPEAKER ADAPTATION APPROACHES : ADAPTATION WITH *K-means all* AND ADAPTATION WITH LVQ CODEBOOKS

Adaptation method		Without Adaptation	Adaptation width *K-means all*	Adaptation with LVQ1	Adaptation with LVQ2
Recognition rate(%)	Top1	49.7	89.9	90.3	89.3
	Top2	64.6	96.6	97.0	95.9
VQ distortion ($\times 10^{-1}$)		3.10	2.31	2.43	2.32

Next, we used the *K-means each* codebook for a new speaker instead of the *K-means all* codebook and we also generated the LVQ codebook which *K-means each* codebook is used to initialize. With these codebooks, a speaker adaptation experiment was performed. The performance of the speaker adaptation approach with LVQ codebook and that of the scheme with *K-means each* codebook were compared. We also compared VQ distortion between the test speech and each codebook. The results are shown in Table 2. Comparing Table 1 with Table 2, respectively, we can find that adaptation with *K-means each* codebook results in higher distortion error than that with *K-means all* codebook, but the recognition rate is better, and that LVQ2 codebook, in which *K-means each* codebook is used to initialize, yields the best recognition rate.

TABLE 2
PERFORMANCE COMPARISON OF SPEAKER ADAPTATION APPROACHES : ADAPTATION WITH *K-means each* AND ADAPTATION WITH LVQ CODEBOOKS

Adaptation method		Adaptation width *K-means each*	Adaptation with LVQ1	Adaptation with LVQ2
Recognition rate(%)	Top1	90.6	89.9	90.0
	Top2	97.1	96.6	97.2
VQ distortion ($X\ 10^{-1}$)		2.42	2.45	2.43

We also made another speaker adaptation experiment based on CT and LVQ. First, we made a speaker adaptation system based on LVQ algorithm in which the optimal path alignment between the new speaker and the reference is found and the LVQ codebook is obtained from the adaptation speech labeled by this path alignment. At this time, however, two codebooks whose domains are cepstral coefficients and their differences, are replaced by two LVQ codebooks, respectively. We empirically found that replacing two codebooks by their LVQ codebooks, gives higher recognition rate than replacing only one codebook. With this adaptation system, we implemented the modified CT algorithm as a post-processor of this system.

According to the variation of the CT iteration number, we compared the performance of the speaker adaptation method with the LVQ codebook to that of the scheme with the modified CT algorithm in which the maximum number of iteration is set to be 2 and the between-class learning rate is set to be 1. The results are shown in Figure 2. In this figure, the average recognition rate of three persons are drawn. It can be seen that the performances of all speaker adaptation systems are improved after adopting the modified CT algorithm, and that the highest recognition rate is obtained when the modified CT algorithm is used as a post-processor of the speaker adaptation based on LVQ1 codebook. Table 3 shows the best result of each speaker adaptation system in detail. The result shows the best recognition rate of 92.8 % for the top 1, and 97.4 % for the top 2, when the modified CT algorithm was performed on the speaker adaptation system based on LVQ1 codebook in which *K-means all* codebook is used to initialize the LVQ1 codebook.

TABLE 3
PERFORMANCE COMPARISON OF SPEAKER ADAPTATION SYSTEM BASED ON CT AND LVQ

(a) Reference → male 1 (b) Reference → male 2
(c) Reference → female (d) Average

Adaptation method	Recognition rate(%)							
	(a)		(b)		(c)		(d)	
	Top 1	Top2	Top1	Top2	Top1	Top2	Top1	Top2
LVQ1 *(K-means all)*	96.3	100.0	91.0	95.7	91.0	96.7	92.8	97.4
LVQ2 *(K-means all)*	95.3	99.0	90.3	94.7	91.7	97.7	92.4	97.1
LVQ2 *(K-means each)*	95.0	99.3	90.3	97.3	88.3	95.3	91.2	97.3

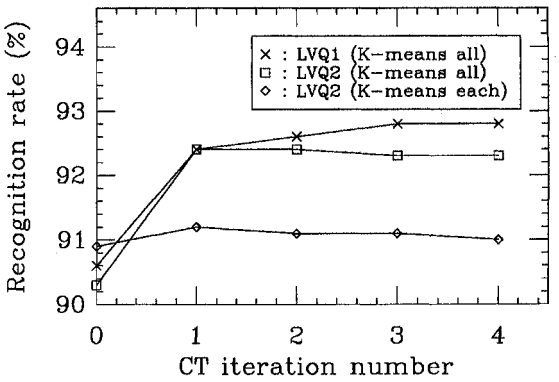

Fig. 2. Average recognition rate of the adaptation system based on CT and LVQ

VI. CONCLUSION

We have presented a speaker adaptation system based on CT and LVQ. Our algorithm consists of two stages: codebook adaptaion and HMM parameter adaptation. In the stage of codebook adaptation, LVQ codebook generated from adaptation speech gave better recognition rate than conventional *K-means* codebook. In the stage of HMM parameter adaptation, the modified corrective training algorithm for speaker adaptation improved the recognition rate. This algorithm was implemented in a speaker adaptive system as a post-processor of the spectral mapping algorithm. A baseline system with the recognition rate of 96.3% was first established. With this, we compared the performance of the speaker adaptation approach with the conventional algorithm to that of the scheme with the proposed algorithm. The results showed that the best recognition rate was obtained when the modified CT algorithm was performed on the speaker adaptation system based on LVQ1 codebook in which the *K-means all* codebook is used to initialize the LVQ1.

References

[1] M. Feng. "Iterative normalization for speaker-adaptive training in continuous speech recognition," ICASSP, Paper S12.4, 1989.

[2] T. Kohonen, et al. "Statistical pattern recognition with neural networks: benchmarking studies," IEEE, Proc. of ICNN, pp.61-68, 1988

[3] T. H. Applebaum and B. A. Hanson. "Enhancing the discrimination of speaker independent hidden Markov model with corrective training", Proc. ICASSP, 1989, Paper S6.13

[4] K. F. Lee and H. W. Hon. "Speaker-independent phone recognition using hidden Markov models", IEEE Trans., 1989, ASSP-37, pp.1641-1648