Discriminative Transformations of Speech Features Based on Minimum Classification Error

Behzad Zamani[1], Ahmad Akbari[1], Babak Nasersharif[1, 2], Azarakhsh Jalalvand[1,3]
[1]Audio & Speech Processing Lab, Iran University of Science & Technology, Iran
[2]Computer Engineering Department, University of Guilan, Iran
[3]Electronics and Information Systems (ELIS) department, Ghent University, Belgium
{bzamani, akbari, nasser_s, jalalvand}@iust.ac.ir

Abstract: *Feature extraction is an important step in pattern classification and speech recognition. Extracted features should discriminate classes from each other while being robust to the environmental conditions such as noise. For this purpose, some transformations are applied to features. In this paper, we propose a framework to improve independent feature transformations such as PCA (Principal Component Analysis), and HLDA (Heteroscedastic LDA) using the minimum classification error criterion. In this method, we modify full transformation matrices such that classification error is minimized for mapped features. We don't reduce feature vector dimension in this mapping. The proposed methods are evaluated for continuous phoneme recognition on clean and noisy TIMIT. Experimental results show that our proposed methods improve performance of PCA, and HLDA transformation for MFCC in both clean and noisy conditions***.**

Keywords: Feature transformation, Minimum classification error, Speech recognition.

1. Introduction

Speech recognition systems include two main components: feature extraction and classification. The classification module is usually designed using statistical approach. A well-known classification method for speech recognition is Hidden Markov Models (HMMs). The HMM parameters are trained automatically using a training data set. A conventional training algorithm is the expectation maximization algorithm using the ML criterion. This algorithm only maximizes likelihood of each individual class. It is not chosen to discriminate between classes. Thus, discriminative training methods such as minimum classification error (MCE) [1] and maximum mutual information (MMI) [2] have been proposed.

In the feature extraction module, the useful discriminative information is extracted from speech signal such that the HMM classifier can recognize different speech units including phones, tri-phones, syllables or words. The most widely used and successful speech features are Mel-frequency cepstral coefficients (MFCC). However, MFCC are not optimal for discriminating of speech units. Their performance degrades in the presence of additive noise. Hence, several methods have been suggested for extracting more discriminative and robust features [3][4]. Linear Discriminant Analysis (LDA) based transformation is one of these approaches which transform the standard MFCCs into more discriminative features [4].

LDA transformation and its family such as Heteroscedastic LDA (HLDA) [5] and kernel LDA (KLDA) [6] can be used instead of DCT or along with DCT in MFCC extraction process [7]. Transformations like LDA and Principal Component Analysis (PCA) project the original feature vectors into a new feature space through a linear transformation matrix. They optimize transformation matrix with different goals. PCA optimizes the transformation matrix by finding the largest variations in the original feature space. On the other hand, LDA maximizes ratio of between-class variation and within-class variation when projecting the features into a subspace.

The drawback of these transformations is that their optimization criteria are different from the classifier's minimum classification error criterion [8], and can potentially corrupt the classifier performance. There are several methods to overcome this drawback. In some approaches, feature extraction and classification are conducted jointly based on a consistent criterion [4]. Minimum Classification Error (MCE) training method is an example of such methods.

In this paper, we propose a framework to improve feature transformation matrices (such as PCA and HLDA) using the MCE criterion. The framework provides a full transformation matrix for mapping features. Mapping is performed without any feature dimension reduction.

The rest of this paper is organized as follows. Section 2 includes a brief introduction of the MCE method. Section 3 explains our proposed framework for optimizing of the feature transformation matrix. Section 4 contains our experimental results reported on TIMIT database. Finally, we give our conclusion in Section 5.

2. Minimum Classification Error

Minimum classification error is a well-known discriminative method used for both feature transformation [9] and classifier training [1]. When the MCE method is used in training of the HMM, the parameters are adjusted to reduce the total classification error [1]. In the MCE training method, the objective function to find the HMM parameters is modeled first using a continuous function. Then, minimum of the function is found using a gradient-search method such as gradient probabilistic descent (GPD) technique [1]. However, the gradient search approaches often get trapped in local optima. A genetic-based MCE algorithm is proposed in [10] to overcome this problem.

The main idea behind the MCE algorithm is to optimize an empirical error rate on the training set to improve the overall recognition rate. After the empirical training error rate is optimized by a classifier or recognizer, a biased estimate of the true error rate is obtained. One effective way to reduce this bias rate is to increase "margins" on the training data. It is desirable to use such large margins for achieving lower test errors, even if this may result in higher empirical errors in the training. This leads to methods such as Large-Margin MCE (LM-MCE) which adjusts the margin incrementally in the MCE training process such that a desirable balance can be achieved between the empirical error rates on the training set and the margin [11].

De La Torre used MCE for finding a feature transformation matrix that minimizes classification error [9]. After that, Wang and Paliwal modified this method for vowel recognition [8]. These methods define two cost functions. The first one based on addition (indexed Add) as in Equation (1) [9]. The second one based on division (indexed Div) as in Equation (2) [8].

$$d_{k,Add}(O_{n_k},F) = -g_k(O_{n_k},F) + \frac{1}{\eta}\log\left[\frac{1}{I-1}\sum_{i=1,i\neq k}^{I}\exp\left(g_i(O_{n_k},F)\eta\right)\right] \tag{1}$$

$$d_{k,Div}(O_{n_k},F) = \frac{\log\left[\frac{1}{I-1}\sum_{i=1,i\neq k}^{I}\exp\left(g_i(O_{n_k},F)\eta\right)\right]}{\eta g_k(O_{n_k},F)} \tag{2}$$

where, F stands for the feature extraction parameters, I is the number of classes, η is cluster weighting which is tuned empirically. H is a positive number represented in [1]. H is a small fractional value that is necessary due to the very small log probabilities. $g_i(o_{n_k},F)$ is generated logarithm likelihood for observation O_{n_k} defined as:

$$g_i\left(O_{n_k},F\right) = \log p\left(O_{n_k},Q;\lambda_i\right) = \log\left(\pi_{q_0}^{(i)}\prod_{t=1}^{T}a_{q_{t-1}q_t}^{(i)}b_{q_t}^{(i)}\left(o_{n_k,t}\right)\right) =$$
$$\log\pi_{q_0}^{(i)} + \sum_{t=1}^{T}\log a_{q_{t-1}q_t}^{(i)} + \sum_{t=1}^{T}\log b_{q_t}^{(i)}\left(o_{n_k,t}\right)$$
$$O_{n_k} = W \times X_{n_k} \tag{3}$$

where, o_{n_k} is n_k-th transformed observations sequence in class k shown by $O_{n_k} = \{o_{n_k,1}, o_{n_k,2}, \ldots, o_{n_k,T}\}$, X_{n_k} is n_k-th original observation sequence in class k as $X_{n_k} = \{x_{n_k,1}, x_{n_k,2}, \ldots, x_{n_k,T}\}$, Q is optimal HMM state sequence shown by $Q = \{q_0, q_1, q_2, \ldots, q_T\}$ that achieves maximum of $p\left(O_{n_k};\lambda_i\right)$. $p\left(O_{n_k},Q;\lambda_i\right)$ is generated likelihood for o_{n_k} by Hidden Markov Model λ_i with optimal state sequence Q. $\pi_{q_0}^{(i)}$ is the initial state probability of the i-th HMM, $a_{q_{t-1}q_t}^{(i)}$ is the probability of making a transition from state q_{t-1} to state q_t for the i-th HMM and $b_j^{(i)}\left(o_{n_k,t}\right)$ is generated probability for observing vector $o_{n_k,t}$ in the j-th state of the i-th HMM.

W is transformation matrix to be determined using MCE method. In the MCE methods, our object is to minimize the misclassification measures (1) and (2). Thus, we define a cost function that maps the misclassification measure between zero and one. For this purpose, we select sigmoid function as the cost function [8] [12]:

$$l_k(O_{n_k},F) = \frac{1}{1+\exp\left(-\alpha\, d_k(O_{n_k},F)\right)} \tag{4}$$

Where, $d_k(o_{n_k},F)$ is as in (1) and (2), and α is a tuning parameter greater than one. Obviously, when $d_{k,Add}(o_{n_k},F)$ is smaller than zero in equation (1), implying an accurate classification, $l_k(o_{n_k},F)$ will be close to zero indicating no loss. A positive $d_{k,Add}(o_{n_k},F)$ represents a penalty in the form of the classification error. For an observation, o_{n_k}, the total classification error is computed by [8,9]:

$$L = \sum_{k=1}^{I}\sum_{n_k=1}^{N_k} l_k(O_{n_k},F) \tag{5}$$

We want to find the transformation matrix W which minimizes total classification error, L. The transformation matrix can now be computed using the gradient descent method for function L as represented in [9]:

$$w_{n,m,iter} = w_{n,m,iter-1} - \beta\frac{\partial L}{\partial w_n} \tag{6}$$

Or in a matrix form as,

$$W_{iter} = W_{iter-1} - \beta \frac{\partial L}{\partial W} \qquad (7)$$

Where, W is the transformation matrix; n and m indicate indexes of elements of W; *iter* denotes iteration number in gradient descent algorithm and β is the learning parameter. Equations (6) and (7) represent an iterative process. The iteration is stopped when total classification error, L, is lower than a threshold. In this paper, we use full transformation MCE matrix in relation (7).

3. Improving Feature Transformation Using MCE Method

One of the drawbacks of the LDA and PCA transformations is that their optimization criteria are different from the minimum classification error criteria of the classifier which may distort the classifier performance. We propose a framework here to optimize such matrices using the MCE criteria. Suppose that the transformation matrix W transforms the original n dimensional feature vector x into a new n dimensional vector y. In fact, elements of transformation matrix W are the parameters set of the feature extraction module. We want to compute a feature transformation matrix W which minimizes a cost function of classification error. For this purpose, we compute the derivative of relation (4) with respect to W [7]:

$$\frac{\partial l_k(O_{n_k},F)}{\partial W} = \frac{\partial l_k(O_{n_k},F)}{\partial d_k(O_{n_k},F)} \frac{\partial d_k(O_{n_k},F)}{\partial W} \qquad (8)$$

Where, cost function $l_k(o_{n_k},F)$ is defined as a sigmoid function of an error measure, $d_k(o_{n_k},F)$, as in (4). Using relation (4), we obtain,

$$\frac{\partial l_k(O_{n_k},F)}{\partial d_k(O_{n_k},F)} = \alpha l_k(O_{n_k},F)\left(1 - l_k(O_{n_k},F)\right) \qquad (9)$$

To compute the second term $\frac{\partial d_k(o_{n_k},F)}{\partial W}$ in relation (8), we use relations (1) and (2). After computing derivations of relations (1) and (2), we can write:

$$\frac{\partial d_{k,Add}(O_{n_k},F)}{\partial W} = -\frac{\partial g_k(O_{n_k},F)}{\partial W} + \sum_{i=1,i\neq k}^{I} \left\{ \frac{\exp\left(g_i(O_{n_k},F)\eta\right)}{\sum_{j=1,j\neq k}^{I} \exp\left(g_j(O_{n_k},F)\eta\right)} \times \frac{\partial g_i(O_{n_k},F)}{\partial W} \right\} \qquad (10)$$

$$\frac{\partial d_{k,Div}(O_{n_k},F)}{\partial W} = \eta^{-1} g_k^{-1}(O_{n_k},F) \sum_{i=1,i\neq k}^{I} \left\{ \frac{\exp\left(g_i(O_{n_k},F)\eta\right)}{\sum_{j=1,j\neq k}^{I} \exp\left(g_j(O_{n_k},F)\eta\right)} \times \frac{\partial g_i(O_{n_k},F)}{\partial W} \right\} - \frac{\partial g_k(O_{n_k},F)}{\partial W} g_k^{-2}(O_{n_k},F) \log\left[\frac{1}{I-1} \sum_{i=1,i\neq k}^{I} \exp\left(g_i(O_{n_k},F)\eta\right)\right]^{1/\eta^2}$$

(11)

Now, we need to compute $\frac{\partial g_i(o_{n_k},F)}{\partial W}$ in relation (10) and (11). Since, we can obtain $\frac{\partial g_i(o_{n_k},F)}{\partial W}$ by differentiating of relation (3) with respect to W. The derivation result can be written as relation (12). In relation (3), term $b_j^{(i)}\left(o_{n_k,t}\right)$ is a function of o_{n_k} and so W (because $O_{n_k} = W \times X_{n_k}$). $\pi_{q_0}^{(i)}$ and $a_{q_{t-1}q_t}^{(i)}$ are constant respect to O_{n_k} and so W.

$$\frac{\partial g_k(O_{n_k},F)}{\partial W} = \sum_{t=1}^{T} \delta(q_t - j) \frac{1}{b_j^{(k)}\left(o_{n_k,t}\right)} \frac{\partial b_j^{(k)}\left(o_{n_k,t}\right)}{\partial W} = -\sum_{t=1}^{T} \delta(q_t - j) \sum_{m=1}^{M} \gamma_{jm}^{(k)}\left(o_{n_k,t}\right) \left[\left(C_{jm}^{(k)}\right)^{-1} \left(o_{n_k,t} - \mu_{jm}^{(k)}\right) x_t^T\right]$$

(12)

Where, $k = 1, 2, \ldots, I$ denotes the I competing models; j is the current state index; M is number of mixtures; $\delta(.)$ is *Kronecker* delta function; q_t is HMM state at *time t*; x_t is the t-th original feature vector; $o_{n_k,t}$ indicates transformed feature vector using $o_{n_k,t} = wx_{n_k,t}$; T is the number of observations and $b_j^{(k)}\left(o_{n_k,t}\right)$ is generated probability for observing vector $o_{n_k,t}$ in the j-th state of the k-th HMM, defined as:

$$b_j^{(k)}\left(o_{n_k,t}\right) = \sum_{m=1}^{M} c_{jm}^{(k)} b_{jm}^{(k)}\left(o_{n_k,t}\right) = \frac{1}{(2\pi)^{n/2}} \sum_{m=1}^{M} \frac{c_{jm}^{(k)}}{\left|C_{jm}^{(k)}\right|^{1/2}} \exp\left(-\frac{1}{2}\left(o_{n_k,t} - \mu_{jm}^{(k)}\right)^T \left(C_{jm}^{(k)}\right)^{-1} \left(o_{n_k,t} - \mu_{jm}^{(k)}\right)\right)$$

(13)

Where,

$o_{n_k,t}$: Observation vector for frame t

$\mu_{jm}^{(k)}$: Mean vector of the m-th Gaussian mixture in the state j of the k-th HMM

n: dimensionality of observation vector

$C_{jm}^{(k)}$: Covariance matrix of the m-th Gaussian mixture in state j of the k-th HMM

$c_{jm}^{(k)}$: Weight of the m-th Gaussian mixture in the state j of the k-th HMM *and* $\gamma_{jm}^{(k)}\left(o_{n_k,t}\right)$ is defined as:

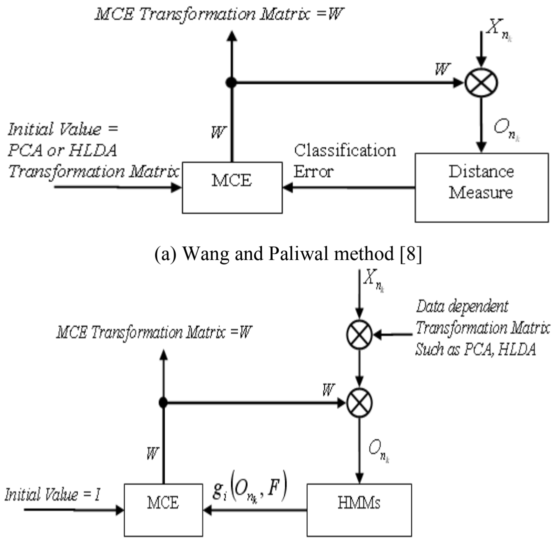
$$\gamma_{jm}^{(k)}\left(o_{n_k,t}\right) = \frac{c_{jm}^{(k)} b_{jm}^{(k)}\left(o_{n_k,t}\right)}{b_j^{(k)}\left(o_{n_k,t}\right)}$$

(14)

Where, $b_{jm}^{(k)}(o_{n_k,t})$ is generated probability by *m*-th Gaussian mixture for observing vector $o_{n_k,t}$ in the state *j* of the *k*-th HMM.

By inserting (13) and (14) in (12) and then by inserting (12) in each of equations (10) and (11), we can compute $\frac{\partial d_k(o_{n_k},F)}{\partial W}$. Then, replacing the computed derivatives in (8) and (9) and in (7), we compute the transformation matrix. In this paper, we use HLDA and PCA transformation matrices as *W***,** and then, optimize the feature transformation matrix *W* using the formula in (7). The proposed method is different from the method proposed in [8].

Fig. 1 shows this difference. Our method can be represented as minimizing classification error in a mapped space that provides optimized mapping matrix. It provides the MCE matrix for a mapped space and so mapped features, while the other proposed method finds the MCE matrix for the original space and so original features. It should be noticed that after obtaining our improved transformation matrix, we apply it to the original features in both test and train phases. In addition, Wang and Paliwal used distance classifier to calculate total classification error, while we used likelihoods of HMM.


MCE Transformation Matrix =W
X_{n_k}
W
O_{n_k}
W
Initial Value = PCA or HLDA Transformation Matrix
MCE
Classification Error
Distance Measure
X_{n_k}
Data dependent Transformation Matrix Such as PCA, HLDA
MCE Transformation Matrix =W
W
O_{n_k}
W
Initial Value = I
MCE
$g_i(O_{n_k},F)$
HMMs

(a) Wang and Paliwal method [8]

(b) The proposed method

Fig. 1. Differences between the proposed method and Wang and Paliwal method

4. Improving Feature Transformation Using MCE Method

We have introduced our abbreviations for the names of methods in Table. I. “Imp” in the name indicates that this method computes an improved transformation matrix on mapped features, while “Init” indicates that it computes transformation matrix using the original (initial) features.

To evaluate the proposed methods, we have performed several experiments on the TIMIT phone recognition task. We use 39 phone classes as in [13]. We report phoneme error recognition rate (PER) on TIMIT database. In all experiments, we use 39-dimension feature vectors consisting of energy, 12 MFCCs and their first and second order derivatives. The features were normalized to have mean zero and standard deviation one over TIMIT training set.

TABLE I. Explanation of Algorithm names (*: proposed methods)

MCE-D	MCE Algorithm using relation (2)
MCE-A	MCE Algorithm using relation (1)
PCA	Principal component analysis
HLDA	Heteroscedastic LDA [5]
PCAMCE-A-Init	PCA matrix as initial value for MCE-A
PCAMCE-D-Init	PCA matrix as initial value for MCE-D
***PCAMCE-A-Imp**	Improved feature transformation using PCA and MCE-A method
***PCAMCE-D-Imp**	Improved feature transformation using PCA and MCE-D
HLDAMCE-A-Init	HLDA matrix as initial value for MCE-A
HLDAMCE-D-Init	HLDA matrix as initial value for MCE-D
***HLDAMCE-A-Imp**	Improved feature transformation using HLDA and MCE-A
***HLDAMCE-D-Imp**	Improved feature transformation method

We don’t reduce features vector dimension in our methods. In addition, we use HMMs with 3 states and 16 Gaussian mixtures per state. We use TIMIT train set for training HMMs and utilize its test set for recognition experiments. For showing performance of our methods in noisy conditions, we added noise to TIMIT test set. We selected three noises from NOISEX92 database: white, pink and factory1. Then, we added these noises to all of TIMIT test set sentences with different SNR values of 0, 5, 10, 15 and 20 dB. HMMs are trained using clean training set of TIMIT database. MCE parameters, α, β and η, for TIMIT, have been set to 0.1, 0.1 and 0.005, respectively.

TABLE II. PER on clean TIMIT noisy and clean test sets

	Clean	**Noisy (Average on 0-20dB and 3 noise types)**
MFCC	*28.31*	*52.08*
MCE-A	*28.17*	*52.03*
MCE-D	*28.24*	*52.01*
PCA	*29.82*	*52.49*
PCAMCE-D-Imp	*28.21*	*51.94*
PCAMCE-A-Imp	***28.19***	***51.83***
PCAMCE-D-Init	*28.75*	*52.32*
PCAMCE-A-Init	*31.01*	*54.01*
HLDA	*29.61*	*51.23*
HLDAMCE-D- Imp	*28.17*	*51.21*
HLDAMCE-A- Imp	***28.07***	***51.12***
HLDAMCE-D-Init	*28.78*	*52.30*
HLDAMCE-A-Init	*29.71*	*51.93*

In Table II, we report phone error recognition results for noisy and clean test set. Results on noisy test set are averaged on all three noise types and SNR values 0 to 20 dB. As shown in the table, the proposed framework improves the performance of PCA and HLDA transformation in both noisy and clean conditions. In addition, improved transformations also do better than Wang's methods (named by "init"). Improved HLDA transformation has the best results among other methods.

5. Conclusions

In this paper, we proposed a framework for improving PCA and HLDA feature transformation methods based on the minimum classification error criterion. In our approach, we change full transformation matrices such that the classification error is minimized for mapped features. This can also be considered as minimizing the classification error in a mapped space generated by improved mapping matrices. In our method, the dimension of original and the mapped spaces are the same. These optimized matrices improve PCA and HLDA performance on noisy and clean TIMIT database for continuous phoneme recognition.

References

[1] B. H. Juang, W.Chou and C.H.Lee, "Minimum classification error rate methods for speech recognition", *IEEE Transaction on Speech and Audio Processing,* Vol. 5, No. 3, pp. 257-265, 1997.

[2] V. Valtchev, *Discriminative methods in HMM-based speech recognition*, P.h.D thesis, Cambridge University, Cambridge, UK, 1995.

[3] B. Nasersharif and A. Akbari, "SNR-dependent compression of enhanced Mel sub-band energies for compensation of noise effects on MFCC features", *Pattern Recognition Letters*, Vol. 28, No. 11, pp. 1320-1326, 2007.

[4] P. Somervuo, "Experiments With Linear And Nonlinear Feature Transformations In HMM Based Phone Recognition", *IEEE International conference on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 52-55, 2003.

[5] M. Loog and R. P. W. Duin, "Linear Dimensionality Reduction via a Heteroscedastic Extension of LDA: The Chernoff Criterion", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 6, pp.732-739, 2004.

[6] A. Kocsor and L.Toth, "Kernel-Based Feature Extraction with a Speech Technology Application", *IEEE Transaction on Signal Processing*, Vol. 52, No. 8, pp. 2250-2263. 2004.

[7] X. B.Li , J. Y.Li, and R. H. Wang, "Dimensionality reduction using MCE-optimized LDA transformation", *IEEE International conference on Acoustics, Speech and Signal Processing*, pp. 137-140, 2004..

[8] X.Wang, and K. K. Paliwal, "Feature extraction and dimensionality reduction algorithms and their applications in vowel recognition", *Pattern Recognition*, Vol. 36, pp. 2429-2439, 2003.

[9] A. De La Torre, A. M Peinado, A. J. Rubio, V. E. Sanchez and J. E. Diaz, , "An Application of Minimum Classification Error to Feature Space Transformation for Speech Recognition", *Speech Communication*, No. 20, pp. 273-290, 1996.

[10] S. Kwong and Q. H. He, "Minimum classification Error Rate Method using Genetic Algorithms", *Signal Processing*, Vol. 82, No. 5, pp. 737-748, 2002.

[11] D. Yu,, L. Deng, X. He and A. Acero, "Use of incrementally regulated discriminative margins in MCE training for speech recognition", *Interspeech Conference*, pp. 2418-2421, 2006.

[12] N. Thakoor, J. Gao and S. Jung,"Hidden Markov Model-Based Weighted Likelihood Discriminant for 2-D Shape Classification", *IEEE Transactions on Image Processing*, Vol. 16, No. 11, pp. 2707-2719, 2007.

[13] K. F. Lee and H. W. Hon, "Speaker-Independent Phone Recognition Using Hidden Markov Models," *IEEE Transaction on Acoustics, Speech and Signal Processing*, Vol. 37, No. 11, pp. 1641-1648, 1989.