Maximum F1-Score Discriminative Training Criterion for Automatic Mispronunciation Detection

Hao Huang, Haihua Xu, Xianhui Wang, and Wushour Silamu

***Abstract*—We carry out an in-depth investigation on a newly proposed Maximum F1-score Criterion (MFC) discriminative training objective function for Goodness of Pronunciation (GOP) based automatic mispronunciation detection that makes use of Gaussian Mixture Model-hidden Markov model (GMM-HMM) as acoustic models. The formulation of MFC seeks to directly optimize F1-score by converting the non-differentiable F1-score function into a continuous objective function to facilitate optimization. We present model-space training algorithm according to MFC using extended Baum–Welch form like update equations based on the weak-sense auxiliary function method. We then present MFC based feature-space discriminative training. We train a matrix projecting from posteriors of Gaussians to a normal size feature space, and add the projected features to traditional spectral features. Mispronunciation detection experiments show MFC based model-space training and feature-space training are effective in improving F1-score and other commonly used evaluation metrics. It is also shown MFC training in both the feature-space and model-space outperforms either model-space training or feature-space training alone, and is about 11.6% better than the maximum likelihood (ML) trained baseline in terms of F1-score. Further, we review and compare mispronunciation detection results with the use of MFC and some traditional training criteria that minimize word error rate in speech recognition. The experimental analysis and comparison provide useful insight into the correlations between F1-score maximization and optimization of these training criteria.**

***Index Terms*—Automatic mispronunciation detection, computer-assisted language learning, discriminative training, F1-score, feature extraction.**

I. Introduction

COMPUTER assisted language learning that makes use of speech and language technologies has gained a growing interest in the last two decades. Automatic mispronunciation detection, which aims at helping the learner by automatically pinpoint erroneous pronunciations, is one of the most popularly deployed applications. Over the last few decades, there are a great deal of research work on automatic mispronunciation detection and a variety of methods have been proposed. A major approach to mispronunciation detection is based on automatic speech recognition (ASR) technologies. Basically, there are two types of ASR based mispronunciation detection techniques. One uses confidence scores such as posterior probability to measure the correctness of a pronunciation [1], [2], [3], [4]. Goodness Of Pronunciation (GOP) [2] is the most conventionally used measurement of pronunciation quality. The other ASR based method uses a phone recognizer to decode the input waveforms with extended pronunciation networks that include correct and incorrect pronunciations to capture possible error types [5]. The ASR based techniques benefit from their well-defined mathematical framework for GMM-HMM based acoustic modeling and many well-developed toolkits for computation.

An alternative to ASR based approach is to use acoustic phonetic features as front-end and a classifier as a back-end such as [6], [7]. Within this framework, mispronunciation detection can be formulated more suitably as a classification problem and thus more discriminative features and classifiers can be explored. With recent successful attempts in applying deep learning methods to speech recognition, new paradigms that make use of deep belief network (DBN) or deep neural network (DNN) in mispronunciation detection or pronunciation evaluation have also been investigated [8], [9], [10], [11], [12].

In this paper, we use GOP based mispronunciation detection method and use GMM-HMM based acoustic models to compute GOP scores. In this approach, GMM-HMM based acoustic models are often trained with maximum likelihood (ML) criterion. In ASR, discriminative training (DT) of the acoustic models has been widely used and has proved to give significant improvement over traditional ML estimation method. The most commonly used discriminative training methods include minimum classification error (MCE) [13], maximum mutual information (MMI) [14] and minimum phone error (MPE) [15], [16]. In ASR, the performance of a system is often evaluated in terms of Word Error Rate (WER) and the above training criteria seek to reducing empirical WER. However, in mispronunciation detection, system performance evaluation metrics are different from WER. The commonly used metrics include False Rejection (FR, correct pronunciations detected as incorrect), False Acceptances (FA, errors detected as correct), True Acceptance (TA, correct pronunciations detected as correct) and True Rejection (TR, errors detected as incorrect). Some works use Precision and Recall (or Precision and Recall curve) to evaluate system performance. These metrics can be effective, however, there has

Manuscript received August 29, 2014; revised December 07, 2014; accepted February 23, 2015. Date of publication March 06, 2015; date of current version March 16, 2015. This work was supported by the National Natural Science Foundation of China under Grants 61365005 and 60965002. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Shinji Watanabe.

H. Huang, X. Wang, and W. Silamu are with the School of Information Science and Engineering, Xinjiang University, Urumqi 830046, China (e-mail: hwanghao@gmail.com; huanghao@xju.edu.cn; wisdom@xju.edu.cn; wushour@xju.edu.cn).

H. Xu is with the Temasek Laboratories, Nanyang Technological University, Singapore, 637553 (e-mail:haihuaxu@ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2015.2409733

often to be an empirical tradeoff among the multiple objectives when they are used as training objectives.

The F1-score, a synthetic one-dimensional indicator, is nowadays an important metric when evaluating the performance of a natural language processing (NLP) system or an information retrieval (IR) system. Recently, researchers began to refine system parameters by directly maximizing the F1-score [17], [18], [19] for logistic regression based classifiers in NLP. In mispronunciation detection, people have began to use F1-score to evaluate system performance [9], [20], [21]. However, few methods that directly optimize acoustic models in terms of the maximization of F1-score, despite its popularity, have been addressed so far. Inspired by these, we proposed a discriminative training criterion which aims at maximizing the empirical F1-score on the annotated non-native speech data for GMM-HMM based automatic mispronunciation [22]. The training objective function is a smoothed form of F1-score function, denoted as Maximum F1-score Criterion (MFC). Extended Baum-Welch (EBW) form like model parameter update equations are derived using the weak-sense auxiliary function method [16]. Mispronunciation detection experiments show MFC based model-space training is effective in improving F1-score on both the training set and the test set.

In speech recognition, the acoustic features are often extracted from the cepstral coefficients using linear transforms, which can be estimated using Linear Discriminant Analysis (LDA) [23], Heteroscedastic Linear Discriminant Analysis (HLDA) [24], or Maximum Likelihood Linear Transform (MLLT) [25] and LDA + MLLT combination [26]. However, the training targets of these methods do not correlate well with the recognition errors, hence the features optimized under such criteria are not optimal in terms of minimizing WER. In the past decade, several discriminatively trained feature transform approaches have been successfully used in large vocabulary continuous speech recognition (LVCSR) systems using GMM-HMMs based acoustic modeling. Among them, feature-space minimum phone error (fMPE) [27] and discriminatively trained region-dependent linear transform (RDLT) [28] are two most popular methods. For robust speech recognition, stereo-based piecewise linear compensation for environments (SPLICE) [29], [30] and its MMI based extension [31] have been proposed. As pointed in [32], the methods share in common that the acoustic space is divided into multiple regions through a global GMM, each region is associated with a distinct feature transform or compensation matrix. They also share in common that the transform(s) could be trained according to some discriminative training criterion.

Inspired by these, to show the effectiveness of MFC training from a discriminative feature training perspective, we investigate MFC based region-dependent linear compensation (MFC-RDLC). The method is to train a matrix that projects posteriors of Gaussians to a normal size feature space, and then to add the projected features to traditional spectral features. The matrix is trained according to MFC objective function. Mispronunciation detection results show MFC based feature-space discriminative training is also effective in increasing the F1-score on both the training data and evaluation data. It is also shown MFC feature training followed by MFC model training can obtain further improvements. This suggests that feature training according to a discriminative training criterion, which has been popular in an ASR task, is also applicable to automatic mispronunciation detection.

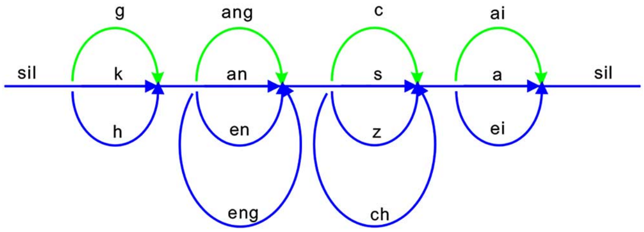

Fig. 1. Example of phone-level confusion network for the word ‘gang1 cai2’, created through concatenating the sublattice of possible errors for each phone, the topmost green arcs correspond to the canonical transcription.

In the final part of the experiments, we carry out a comparative study by showing mispronunciation detection results using acoustic models trained by commonly used discriminative training criteria in ASR such as MMI and MPE. Our major contributions include:

- We set F1-score maximization as the training target for automatic mispronunciation detection. The non-differentiability of F1-score function makes it difficult to maximize directly. We transfer F1-score criterion into a smooth objective function to make it easy to optimize.
- We present model and feature training algorithms according to MFC, showing that they are both effective in enhancing F1-score and other evaluation metrics such as Precision and Recall, etc.
- The empirical comparisons helps to figure out the relationship between the MFC objective function and those commonly used training disciplines that minimize empirical word (or phone) error rate in speech recognition. We hope this analysis and comparison can provide a better understanding of MFC training.

The remainder of this paper is organized as follows. Section II briefly reviews the GOP based mispronunciation detection method and then presents the formulation of the MFC objective function. Section III presents the GMM-HMM parameter optimization algorithms. Section IV describes feature compensation and optimization. The experimental evaluations and detailed analysis on the results are described in Section V. Section VI draws the conclusion of this study and points out our future work.

II. The MFC Objective Function

A. GOP based Mispronunciation Detection

The task of mispronunciation detection is to verify whether the pronunciation of a phone is correct or not. GOP [2] is a score to measure the quality of a pronunciation. In GOP based mispronunciation detection, a phone-level confusion network which includes canonical phone pronunciations and possible mispronunciations needs to be built. This is normally obtained by using forced alignment against the canonical transcription of the prompt utterance, and then all the possible pronunciation realizations are added. Fig. 1 shows the resulting network for a Chinese word “gang1 cai2”. Given a set of acoustic observations of R training utterances: $\mathcal{O}_r, r = 1, \cdots, R$, let $\mathcal{O}_{r,n}$ be the acoustic observations of the nth phonetic segment in utterance

r that is composed of N_r segments, and the canonical label of segment (r, n) is denoted as $q_{r,n}$, the GOP of phone segment (r, n) is calculated as

$$\begin{aligned}\mathrm{GOP}(\mathcal{O}_{r,n}, q_{r,n}) &= \frac{1}{T_{r,n}} \log P(q_{r,n}|\mathcal{O}_{r,n}) \\ &= \frac{1}{T_{r,n}} \log \left(\frac{p(\mathcal{O}_{r,n}|q_{r,n})P(q_{r,n})}{\sum_{q \in Q(r,n)} p(\mathcal{O}_{r,n}|q)P(q)} \right),\end{aligned} \tag{1}$$

where $T_{r,n}$ is the duration (in frames) of the acoustic observation $\mathcal{O}_{n,r}$ in segment (r, n) and $Q(r, n)$ represents all the possible pronunciations of the segment. $q_{r,n}$ is the canonical pronunciation of segment (r, n). After the individual GOP score has been calculated, a threshold τ is applied to decide whether the phone is correctly pronounced or not:

$$\mathrm{GOP}(\mathcal{O}_{r,n}, q_{r,n}) > \tau \begin{cases} \text{Yes} & \text{correct pronunciation} \\ \text{No} & \text{mispronunciation.} \end{cases} \tag{2}$$

τ is a global or phone dependent threshold that can be empirically selected or statistically tuned. We use a modified form of GOP, which differs a little from the modification used in our previous work [22] and appears to give slightly better F1 result in MFC training:

$$G(r, n) = \log \frac{P(\mathcal{O}_{r,n}|q_{r,n})^{\frac{\kappa}{T_{r,n}}}}{\sum_{q \in Q(r,n)} P(\mathcal{O}_{r,n}|q)^{\frac{\kappa}{T_{r,n}}}}, \tag{3}$$

where all the phones are assumed to share the same prior probability as in [2], $0 < \kappa < 1$ is a commonly applied exponential scaling factor in discriminative training to reduce dynamic range of the probabilities.

If we use (3), an error detection measurement for phone segment (r, n) can be defined as

$$d(r, n) = -G(r, n) + \tau. \tag{4}$$

It can be seen that $d(r, n) > 0$ is interpreted as the segment $O(r, n)$ is detected as erroneous and $d(r, n) \leq 0$ as correct. Let $\mathbb{1}(\cdot)$ be the step indicator function:

$$\mathbb{1}(u) = \begin{cases} 1 & if\ u > 0 \\ 0 & if\ u \leq 0 \end{cases}, \tag{5}$$

the GOP based mispronunciation decision rule can be written as:

$$\mathbb{1}(d(r, n)) = \begin{cases} 1 & \text{mispronunciation} \\ 0 & \text{correct pronunciation} \end{cases}. \tag{6}$$

B. The MFC Objective Function

After the GOPs of all the phonemes in the R utterances are calculated and the mispronunciations are detected according to (6), F1-score can be computed to evaluate the performance of the system based on the detection results of the machine and the results of a human evaluator. F1-score is the harmonic mean of *Precision* and *Recall* computed from the number of mispronunciations detected by both the computer and human evaluator. They are defined as

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{7}$$

and

$$Precision = \frac{N_{WW}}{N_D} \times 100\% \tag{8}$$

$$Recall = \frac{N_{WW}}{N_W} \times 100\%. \tag{9}$$

N_{WW} is the number of phones marked as mispronunciations by both the computer and the human evaluator. N_D is the total number of mispronunciations detected by the machine, N_W is the number of mispronunciations judged by the human evaluator. By replacing *Precision* and *Recall* in (7) with (8) and (9), F1-score can be rewritten as

$$F1 = \frac{2N_{WW}}{N_D + N_W}. \tag{10}$$

In terms of the GOP based error decision rule in (6), N_{WW} and N_D can be expressed as:

$$N_{WW} = \sum_{r=1}^{R} \sum_{n=1}^{N_r} \mathbb{1}(d(r, n)) \times E(r, n) \tag{11}$$

$$N_D = \sum_{r=1}^{R} \sum_{n=1}^{N_r} \mathbb{1}(d(r, n)) \tag{12}$$

$E(r, n)$ is the human-annotated result of segment (r, n). $E(r, n) = 1$ is marked as mispronunciation and 0 otherwise.

In (10), N_W is a constant. N_D and N_{WW} are both functions of the system parameters and we can maximize F1-score by tuning these parameters. However, F1-score is not differentiable because of the step indicator function, which make it difficult to optimize by using a gradient based method. To surpass this problem, we use the sigmoid function to transform the step indicator function $\mathbb{1}(\cdot)$ into a continuous function:

$$S(u) = \frac{1}{1 + \exp(-\xi u)} \tag{13}$$

where the constant $\xi > 0$ and $S(u) \to \mathbb{1}(u)$ as $\xi \to \infty$. By replacing the indicator function $\mathbb{1}(\cdot)$ with $S(\cdot)$ in (11) and (12), the smoothed number of errors marked by both the machine and human annotator can be

$$N_{WW}^S = \sum_{r=1}^{R} \sum_{n=1}^{N_r} S(d(r, n))E(r, n). \tag{14}$$

Similarly, the smoothed number of errors detected by the machine can be

$$N_D^S = \sum_{r=1}^{R} \sum_{n=1}^{N_r} S(d(r, n)). \tag{15}$$

By replacing N_{WW} and N_D with N_{WW}^S and N_D^S in F1-score computation (10), we obtain a smooth form of F1-score, denoted as the *maximum F1-score criterion* (MFC), which can be written as follows:

$$\begin{aligned}\mathcal{F}_{\mathrm{MFC}} &= \frac{2N_{WW}^S}{N_D^S + N_W} &(16)\\ &= \frac{2\sum_{r=1}^{R}\sum_{n=1}^{N_r} S(d(r, n))E(r, n)}{\sum_{r=1}^{R}\sum_{n=1}^{N_r} S(d(r, n)) + N_W}. &(17)\end{aligned}$$

From the equation we can see, as N_W remains fixed given the human-annotation results, the MFC objective function can be maximized by simultaneously increasing N^S_{WW} and decreasing N^S_D.

III. Model Space Discriminative Training

Model-space discriminative is to optimize the MFC objective function by updating the GMM-HMM parameters (In this work, only the update of Gaussian means and variances are considered). Various optimization methods for discriminative training of GMM-HMM have been tried, including Generalized Probabilistic Descent (GPD) algorithm for MCE training in [13], Stochastic Gradient Ascent (SGA) and Resilient Propagation (RProp) for MMI training in [33]. In this work, we use the weak-sense auxiliary function method in [16] and EBW update equations which have proved to be successful in MMI, MPE and Boosted MMI (BMMI) [34]. According to [16], the advantage of using weak-sense auxiliary function method is that there is no need to determine the appropriate learning rate, or use second-order statistics. The weak-sense auxiliary function may be selected so that the resulting EBW algorithm has a closed-form solution for the parameter estimation [16].

A. Weak-Sense Auxiliary Function in MFC Update

The weak-sense auxiliary function is a smooth function such that

$$\left.\frac{\partial \mathcal{Q}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\theta}}\right|_{\hat{\boldsymbol{\theta}}} = \left.\frac{\partial \mathcal{F}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\hat{\boldsymbol{\theta}}}, \tag{18}$$

where $\hat{\boldsymbol{\theta}}$ represents the current model parameters and the variable $\boldsymbol{\theta}$ the parameters to be estimated. According to [16], optimizing the objective function can be achieved by optimizing the weak-sense auxiliary function. For MFC objective function, the weak-sense auxiliary function used can be

$$\mathcal{Q}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}}) = \sum_q \left.\frac{\partial \mathcal{F}_{\mathrm{MFC}}}{\partial \log p(\mathcal{O}_q|q)}\right|_{\hat{\boldsymbol{\theta}}} \log p(\mathcal{O}_q|q) + \mathcal{Q}^{\mathrm{s}}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}}), \tag{19}$$

where q is a phone arc within the training confusion networks, $\mathcal{O}_q$ represents the observation sequence in arc q. A smoothing term $\mathcal{Q}^s(\boldsymbol{\theta},\hat{\boldsymbol{\theta}})$ is necessarily added on right-hand side of (19) to ensure concavity of the auxiliary function, and consequently improve stability in optimization. It must satisfy the following constraint to ensure the resulting auxiliary function is still a valid weak-sense auxiliary function

$$\frac{\partial \mathcal{Q}^{\mathrm{s}}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\theta}}|_{\hat{\boldsymbol{\theta}}} = 0. \tag{20}$$

One form of the smoothing function can be

$$\begin{aligned}\mathcal{Q}^{\mathrm{s}}(\boldsymbol{\theta},\hat{\boldsymbol{\theta}}) = &-\frac{1}{2}\sum_{s=1}^{S}\sum_{m=1}^{M_s} D_{sm}\{\log(2\pi\hat{\sigma}_{sm}) \\ &+ \frac{(\mu_{sm}^2+\sigma_{sm}^2) - 2\mu_{sm}\hat{\mu}_{sm} + \hat{\mu}_{sm}^2}{\hat{\sigma}_{sm}^2}\},\end{aligned} \tag{21}$$

where S is the number of states in the HMM set. and M_s is number of Gaussians in state s. $\hat{\mu}_{sm}$ and $\hat{\sigma}_{sm}$ are respectively the current mean and variance for Gaussian m in state s (consider Gaussians with a single dimension for simplicity in this section). μ_{sm} and σ_{sm} are the mean and variance to be updated. D_{sm} is a Gaussian-dependent factor that will be discussed in (31).

B. Update Equations

To maximize the weak-sense auxiliary function, we need to collect two sets of statistics: the occupation-data, sum-of-data and sum-of-square-data of the 'numerator':

$$\beta^{\mathsf{n}}_{sm} = \sum_{t=1}^{T} \max\left(0, \psi^{\mathsf{MFC}}_{sm}(t)\right) \tag{22}$$

$$\chi^{\mathsf{n}}_{sm} = \sum_{t=1}^{T} \max\left(0, \psi^{\mathsf{MFC}}_{sm}(t)\right) o(t) \tag{23}$$

$$Y^{\mathsf{n}}_{sm} = \sum_{t=1}^{T} \max\left(0, \psi^{\mathsf{MFC}}_{sm}(t)\right) o^2(t), \tag{24}$$

where $\psi^{\mathsf{MFC}}_{sm}(t) = \psi^{\mathsf{MFC}}_{q}\psi_{qsm}(t)$ and $\psi_{qsm}(t)$ is the posterior probability of being in state s and Gaussian mixture m of arc q at time t and can be obtained via a forward-backward pass within arc q. The quantity ψ^{MFC}_{q} is defined as

$$\psi^{\mathsf{MFC}}_{q} = \frac{1}{\kappa}\frac{\partial \mathcal{F}_{\mathrm{MFC}}}{\partial \log p(\mathcal{O}_q|q)}, \tag{25}$$

which will be derived in the later part of this section. The statistics of the 'denominator' can be calculated by replacing $\max(0, \psi^{\mathsf{MFC}}_{sm}(t))$ with $\max(0, -\psi^{\mathsf{MFC}}_{sm}(t))$ in (22)–(24):

$$\beta^{\mathsf{d}}_{sm} = \sum_{t=1}^{T} \max\left(0, -\psi^{\mathsf{MFC}}_{sm}(t)\right) \tag{26}$$

$$\chi^{\mathsf{d}}_{sm} = \sum_{t=1}^{T} \max\left(0, -\psi^{\mathsf{MFC}}_{sm}(t)\right) o(t) \tag{27}$$

$$Y^{\mathsf{d}}_{sm} = \sum_{t=1}^{T} \max\left(0, -\psi^{\mathsf{MFC}}_{sm}(t)\right) o^2(t). \tag{28}$$

When the sufficient statistics are available, analogous to MPE update, maximizing the auxiliary function with respect to the means and variances yields the following update formulae:

$$\mu_{sm} = \frac{\chi^{\mathsf{n}}_{sm} - \chi^{\mathsf{d}}_{sm} + D_{sm}\hat{\mu}_{sm}}{\beta^{\mathsf{n}}_{sm} - \beta^{\mathsf{d}}_{sm} + D_{sm}} \tag{29}$$

$$\sigma^2_{sm} = \frac{Y^{\mathsf{n}}_{sm} - Y^{\mathsf{d}}_{sm} + D_{sm}(\hat{\sigma}^2_{sm} + \hat{\mu}^2_{sm})}{\beta^{\mathsf{n}}_{sm} - \beta^{\mathsf{d}}_{sm} + D_{sm}} - \mu^2_{sm}. \tag{30}$$

The smoothing factor D_{sm} is empirically determined for each Gaussian component and typically set to

$$D_{sm} = E\beta^{\mathsf{d}}_{sm}, \tag{31}$$

where E is a global constant controlling the update speed. A large value of E will slow down the convergence speed. Typically a constant $E = 3.0$ was chosen in the experiments. An important definition for updating the MFC objective function is ψ^{MFC}_{q} in (25). According to the chain rule, the derivative needed to compute this term is written as

$$\begin{aligned}\frac{\partial \mathcal{F}_{\mathrm{MFC}}}{\partial \log p(\mathcal{O}_{r,n}|q)} = &\frac{2}{N^S_D + N_W}\frac{\partial N^S_{WW}}{\partial \log p(\mathcal{O}_{r,n}|q)} \\ &- \frac{2N^S_{WW}}{(N^S_D + N_W)^2}\frac{\partial N^S_D}{\partial \log p(\mathcal{O}_{r,n}|q)}.\end{aligned} \tag{32}$$

The partial derivatives in (32) are written as

$$\frac{\partial N^S_{WW}}{\partial \log p(\mathcal{O}_{r,n}|q)} = \frac{1}{T_{r,n}} E(r,n) \times P \tag{33}$$

$$\frac{\partial N^S_{D}}{\partial \log p(\mathcal{O}_{r,n}|q)} = \frac{1}{T_{r,n}} \times P \tag{34}$$

where

$$P = \frac{\kappa\theta e^{-\theta d(r,n)}}{\left(1+e^{-\theta d(r,n)}\right)^2} \left(\psi_q(r,n) - A(q, q_{r,n})\right). \tag{35}$$

$A(q, q_{r,n})$ is 'phone accuracy' of phone q and is equal to 1 when q and $q_{r,n}$ are the same, and 0 otherwise. $\psi_q(r,n)$ is computed as:

$$\psi_q(r,n) = \frac{P(\mathcal{O}_{r,n}|q)^{\frac{\kappa}{T_{r,n}}}}{\sum_{q' \in Q(r,n)} P(\mathcal{O}_{r,n}|q')^{\frac{\kappa}{T_{r,n}}}}. \tag{36}$$

For clarity, we summarize the iterative MFC model training procedure as follows:

1) Initialize the acoustic models using ML estimation;
2) Iterative MFC model training:
   a) Compute GOP scores of all the phone segments (r,n) in the training utterances;
   b) Search for the best phone-dependent thresholds $\boldsymbol{\tau}$ that maximize $\mathcal{F}_{\text{MFC}}$;
   c) Compute N^S_{WW} and N^S_D using the GOP scores and thresholds obtained in 2.a and 2.b;
   d) Do forward-backward computations to accumulate sufficient statistics in (22)-(28);
   e) Update the means and variances of Gaussians using (29) and (30);
   f) Goto step 2.a unless convergence or maximum number of iterations is reached.

More details can be found in the section of the experiments.

IV. Feature Space Discriminative Training

A. Region Dependent Linear Compensation

Region-dependent feature compensation uses a global Gaussian mixture model to divide the acoustic space into multiple regions, each having a different compensation offset. The output feature of RDLC is the summation of original features with the weighted average of the region-specific offsets, defined as

$$\mathbf{y}_t = \mathbf{o}_t + \sum_{m=1}^{\mathcal{M}} \phi_t^{(m)} \mathbf{b}_m, \tag{37}$$

with $\mathbf{o}_t$ the observation vector of input features at time t, $\mathcal{M}$ being the total number of the Gaussians in the HMM set, $\phi_t^{(m)}$ being the posterior probability of Gaussian given $\mathbf{o}_t$, and $\mathbf{b}_m$ being a region-dependent offset vector of Gaussian m. By defining an offset matrix $\mathbf{B} = [\mathbf{b}_1 \mathbf{b}_2 \cdots \mathbf{b}_m \cdots \mathbf{b}_{\mathcal{M}}]^\top$, and let $\mathbf{\Phi}_t$ be an $\mathcal{M}$-dimensional vector whose elements consist of posterior probabilities $\phi_t^{(m)}$ where $m = 1, \ldots, \mathcal{M}$, RDLC can also be represented as

$$\mathbf{y}_t = \mathbf{o}_t + \mathbf{B}\mathbf{\Phi}_t. \tag{38}$$

Initializing $\mathbf{B}$ zero gives a reasonable starting point for training, i.e., the original spectral features.

B. Optimization of Matrix

We use a gradient ascent algorithm to update the transform matrix $\mathbf{B}$. The derivative of $\mathcal{F}_{\text{MFC}}$ with respect to $\mathbf{B}$ needs to be computed according to the chain rule:

$$\frac{\partial \mathcal{F}_{\text{MFC}}}{\partial \mathbf{B}} = \sum_{t=1}^{T} \frac{\partial \mathcal{F}_{\text{MFC}}}{\partial \mathbf{y}_t} \frac{\partial \mathbf{y}_t}{\partial \mathbf{B}} \tag{39}$$

$$= \sum_{t=1}^{T} \frac{\partial \mathcal{F}_{\text{MFC}}}{\partial \mathbf{y}_t} \mathbf{\Phi}_t^\top. \tag{40}$$

Note that $\mathcal{F}_{\text{MFC}}$ is a function of the features $\mathbf{y}_t (t = 1 \ldots T)$ and the model parameters $\boldsymbol{\theta}$. Moreover, the model parameters $\boldsymbol{\theta}$ $= \{\boldsymbol{\mu}_{sm}, \boldsymbol{\sigma}_{sm}\}$ are also functions of the compensated features $\mathbf{y}_t$. The fact that HMMs will also update during RDLC training also affects the derivative, hence the performance of the optimized feature transform. The chain rule is used to compute the derivative of $\mathcal{F}_{\text{MFC}}$ (17) with respect to the compensated feature at time t:

$$\partial \mathcal{F}_{\text{MFC}} / \partial y_{ti} = P_{ti}^{\text{direct}} + P_{ti}^{\text{indirect}} \tag{41}$$

where y_{ti} is the i-th dimension of the transformed feature vector $\mathbf{y}_t$. The first term on the right-hand side is derived by fixing the model parameters $\boldsymbol{\mu}_{sm}$ and $\boldsymbol{\sigma}_{sm}$ (denoted as 'direct differential' in fMPE [27]) and can be computed using the equation below:

$$P_{ti}^{\text{direct}} = \frac{\kappa}{T_{r,n}} \sum_{s=1}^{S} \sum_{m=1}^{M_s} \psi_{sm}^{\mathsf{MFC}}(t) \frac{\mu_{smi} - y_{ti}}{\sigma_{smi}^2}, \tag{42}$$

where μ_{smi} and σ_{smi}^2 are respectively the i-th dimension of mean $\boldsymbol{\mu}_{sm}$ and variance $\boldsymbol{\sigma}_{sm}^2$. The second term in (41) is referred to as 'indirect differential' in fMPE and can be computed as

$$P_{ti}^{\text{indirect}} = \sum_{s=1}^{S} \sum_{m=1}^{M_s} \left(\frac{\partial \mathcal{F}_{\text{MFC}}}{\partial \mu_{smi}} \frac{\partial \mu_{smi}}{\partial y_{ti}} + \frac{\partial \mathcal{F}_{\text{MFC}}}{\partial \sigma_{smi}^2} \frac{\partial \sigma_{smi}^2}{\partial y_{ti}} \right). \tag{43}$$

The differential of MFC objective function with respect to the Gaussian means and variances in indirect differential can be computed as:

$$\frac{\partial \mathcal{F}_{\text{MFC}}}{\partial \mu_{smi}} = \kappa \mathcal{A}'_{smi} \sigma_{smi}^{-2} \tag{44}$$

$$\frac{\partial \mathcal{F}_{\text{MFC}}}{\partial \sigma_{smi}^2} = \kappa \left(\mathcal{A}''_{smi} \sigma_{smi}^{-4} - \mathcal{A}_{smi} \sigma_{smi}^{-2} \right) / 2 \tag{45}$$

where $\mathcal{A}_{smi}$, $\mathcal{A}'_{smi}$ and $\mathcal{A}''_{smi}$ are Gaussian-dependent statistics that should be accumulated over all the training data in the forward/backward pass within the phone segments:

$$\mathcal{A}_{smi} = \sum_{t=1}^{T} \psi_{sm}^{\mathsf{MFC}}(t) \tag{46}$$

$$\mathcal{A}'_{smi} = \sum_{t=1}^{T} \psi_{sm}^{\mathsf{MFC}}(t) (y_{ti} - \mu_{smi}) \tag{47}$$

$$\mathcal{A}''_{smi} = \sum_{t=1}^{T} \psi_{sm}^{\mathsf{MFC}}(t) (y_{ti} - \mu_{smi})^2. \tag{48}$$

In equation (43), the derivatives of the model parameters with respect to the features, i.e., $\partial \mu_{smi} / \partial y_{ti}$ and $\partial \sigma_{smi} / \partial y_{ti}$ depend on how the models are updated. It is often preferable to assume ML update of the GMM-HMMs rather than discriminative update, so that the optimization will concentrate more on finding discriminative features rather than adapting features to a discriminative model [28]. When the models are updated via ML training, the derivatives of the model with respect to the feature can be computed as follows:

$$\frac{\partial \mu_{smi}}{\partial y_{ti}} = \frac{\psi_{sm}^{\mathsf{ML}}(t)}{\psi_{sm}^{\mathsf{ML}}} \tag{49}$$

$$\frac{\partial \sigma_{smi}}{\partial y_{ti}} = \frac{\psi_{sm}^{\mathsf{ML}}(t)}{\psi_{sm}^{\mathsf{ML}}} \times 2\left(y_{ti} - \mu_{smi}\right) \tag{50}$$

where

$$\psi_{sm}^{\mathsf{ML}} = \sum_{t=1}^{T} \psi_{sm}^{\mathsf{ML}}(t). \tag{51}$$

$\psi_{sm}^{\mathsf{ML}}(t)$ is the posterior probability of Gaussian m at time t given the observations and canonical reference phone sequence in normal ML training. To compute this term, a forward/backward pass on the reference transcripts is needed. We assume that $\psi_{sm}^{\mathsf{ML}}(t)$ is independent of the current acoustic models and use the fixed baseline acoustic models and raw MFCC features to compute this term, because otherwise the derivative would be too complicated.

The derivative $\mathcal{F}_{\mathrm{MFC}}/\mathbf{B}$ is finally fed into the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) routine [35] to update the transform. After the feature transform is updated, the model parameters are updated through single-pass retraining [36] using fixed ML accumulative statistics $\psi_{sm}^{\mathsf{ML}}(t)$. The MFC-RDLC training is also an iterative optimization procedure, as follows:

1) Initialize RDLC matrix by setting $\mathbf{B} \rightarrow \mathbf{0}$;
2) Compute posterior probability vector $\mathbf{\Phi}_t$ using the baseline acoustic models;
3) Adapt the baseline acoustic models on correctly pronounced phone segments in annotated non-native speech data via regular ML training;
4) Using the acoustic models obtained in step 3), do forward-backward computation to get single-pass retraining statistics $\psi_{sm}^{\mathsf{ML}}(t)$ and ψ_{sm}^{ML};
5) Iterative RDLC matrix optimization:
    a) Compute GOP scores for all the phone segments (r, n) in the training utterances;
    b) Search for the best phone-dependent thresholds $\boldsymbol{\tau}$ that maximize $\mathcal{F}_{\mathrm{MFC}}$;
    c) Compute N_{WW}^{S}, N_{WW}^{S} and the $\mathcal{F}_{\mathrm{MFC}}$ using the GOP results and thresholds in 5.a and 5.b;
    d) Do forward-backward computation and calculate accumulative statistics $\mathcal{A}_{smi}$, $\mathcal{A}'_{smi}$ and $\mathcal{A}''_{smi}$ in (46)–(48);
    e) Compute MFC differential with respect to the feature y_{ti} in (41) and hence accumulate $\partial\mathcal{F}_{\mathrm{MFC}}/\partial\mathbf{B}$ against all the time T in (39);
    f) Update $\mathbf{B}$ using L-BFGS and compute new features;
    g) Run single-pass retraining with $\psi_{sm}^{\mathsf{ML}}(t)$ and ψ_{sm}^{ML} in 3) and the new features in 5.f to update the model parameters;
    h) Goto step 5.a unless convergence or maximum number of iterations is reached.

More details can be found in the section of the experiments.

TABLE I
DIVISION OF THE NATIVE (L1) AND NONNATIVE (L2) CORPORA

Data Set	Duration	# Phone Tokens	# Errors
L1 Training	100.6 hr	2,095K	NA
L1 Test	6.9 hr	142K	NA
L2 Training	10.0 hr	98,174	3,077
L2 Test	4.1 hr	39,668	1,022

V. EXPERIMENTS AND RESULTS

A. Databases

The proposed method is evaluated on a Mandarin mispronunciation detection task for Uighur college students who have been learning Putonghua (Mandarin Chinese) in Xinjiang University. In this paper, we focus only on the phone-level mispronunciations without consideration of tonal error detection, which normally belongs to another separate research topic [37], [38]. Therefore, the methods proposed in this paper can also be applied to other non-tonal languages. For the experiments, we have two speech databases:

L1 Database: L1 speech database is the '863 project' Mandarin speech database uttered by native Mandarin speakers. The database contains 92,243 utterances read by 160 speakers (80 male and 80 female). 86,271 utterances out of the database are selected as the L1 training set, which is used to initialize the baseline acoustic models.

L2 Database: L2 database is a non-native speech corpus. The corpus contains speech data collected from 100 Uighur speakers (50 male and 50 female). Each speaker was asked to read three sets of prompted texts. Each set contains 50 single characters, 25 words and 20 short sentences. The database has been annotated by well-trained linguists. After cleaning, the database contains about 14.1 hours of speech, 25,673 utterances. 18,643 utterances of the database are used for MFC training (denoted as L2 training set) and 7,030 are used for mispronunciation detection evaluation (L2 test set). Table I shows the four subsets of the data.

B. Experimental Setup

The construction of the baseline system includes the following steps and the block diagram is shown in Fig. 2.

1) Traditional mono-phone based models are trained using ML estimation on L1 training set. The spectral front-end uses 39 dimensional vector, consisting of 13 MFCCs and their Δ and ΔΔ with cepstral mean normalization. Due to the limited non-native training data, especially the limited amount of mispronounced training data, only monophone HMMs are used. Another reason for using context-independent models is that context information is not easy to implement considering possible mispronunciations around the target phone [6]. The HMM set has 67 phones (28 initials and 37 finals plus silence and short pause), each HMM state is a mixture of 8 Gaussians.
2) Confusion networks are created by using the acoustic models and canonical phone transcriptions. Each segment of the network includes the canonical phone and competitive phone arcs. Phone boundaries of the denominator in GOP computation in (3) could be determined by using

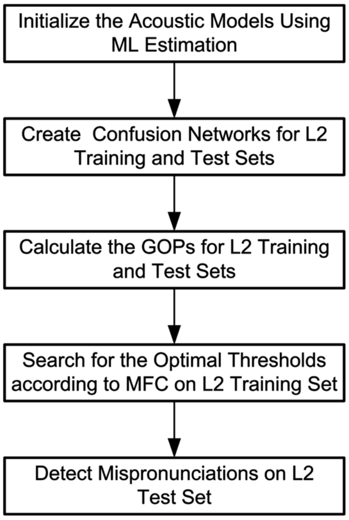
Initialize the Acoustic Models Using ML Estimation
Create Confusion Networks for L2 Training and Test Sets
Calculate the GOPs for L2 Training and Test Sets
Search for the Optimal Thresholds according to MFC on L2 Training Set
Detect Mispronunciations on L2 Test Set

Fig. 2. Flow chart of the baseline system construction.

a phone recognition pass within an unconstrained phone loop network. In our experiments, the phone denominator boundaries were obtained simply by using forced alignment. This is the same as that implemented in [6], which is based on the fact that the boundaries determined from forced alignment are better than those emerging from a phone loop recognition because the speakers try to pronounce as correctly as they can most of the time [6], [20]. For each phone segment (r, n), the denominator phone arcs $Q(r, n)$ can be also predicted by using the output of a phone loop recognizer, or determined by phonological rules designed manually or derived by data driven methods [21], [39]. Here we simply add all the initials into the segment when the canonical phone $q_{r,n}$ is an initial, and add all the finals when $q_{r,n}$ is a final.

3) The acoustic scores in (3) are calculated and then GOP scores of the utterances are obtained.
4) The thresholds $\boldsymbol{\tau}$ in (3) are determined and then mispronunciation detection test is carried out. We use phone dependent thresholds which have previously shown to obtain better performance in [2]. Note that the thresholds are also system parameters and we have already set $\mathcal{F}_{\mathrm{MFC}}$ as the optimization target, it is natural that the thresholds can be obtained by maximizing the MFC objective function on L2 training set:

$$\hat{\boldsymbol{\tau}} = \arg\max_{\boldsymbol{\tau}} \mathcal{F}_{\mathrm{MFC}}. \tag{52}$$

To obtain a closed-form solution to this problem is difficult. We resorted to a search process to find the best $\hat{\boldsymbol{\tau}}$: we tuned the threshold of each phone q using grid search to find a best $\mathcal{F}_{\mathrm{MFC}}$ while maintaining other phone thresholds fixed. The procedure was repeated until the objective function $\mathcal{F}_{\mathrm{MFC}}$ converged to an optimum.

C. Baseline Results

With the ML trained acoustic models and optimized phone-dependent thresholds, the baseline F1-scores on the training and test sets are respectively 0.386 and 0.382, as shown in Table II.

TABLE II
MFC MODEL-SPACE TRAINING RESULTS

	Training		Test	
	$\mathcal{F}_{\mathrm{MFC}}$	F1	$\mathcal{F}_{\mathrm{MFC}}$	F1
Baseline	0.368	0.386	0.367	0.382
+ML Model training (L2)	0.367	0.388	0.363	0.381
+MFC Model training	0.616	0.665	0.454	0.488

In mispronunciation detection, native and non-native databases are commonly recorded in different acoustic environments. In this work, the ML estimated baseline is trained on the native speech database and the MFC training is conducted on the non-native speech database. One might think that the gains of MFC training are due to the adaptation of the baseline acoustic models to L2 acoustic conditions. We added an additional ML training process on the correctly pronounced L2 data using the baseline models as a start point. It is seen F1-scores on the training and test sets are respectively 0.388 and 0.381, that is, no clear F1-score improvement is obtained. This indicates in our experiments the adaption of the baseline acoustic models to L2 acoustic conditions is not helpful in improving mispronunciation detection performance.

D. MFC Model Space Training Results

The MFC model training was carried out on the acoustic models adapted on L2 training data. The probability scaling factor was empirically set to $\kappa = 0.1$ and the sigmoid constant was $\xi = 10.0$. 20 iterations of MFC model training has shown to be sufficient for the objective function to converge.

We added a threshold searching process after model update within each MFC model training iteration. This was denoted as MFC model training with Iterative Threshold Tuning (MFC-ITT) in [22]. As shown in [22], if we only update the model parameters iteratively while remaining the phone thresholds $\boldsymbol{\tau}$ fixed, $\mathcal{F}_{\mathrm{MFC}}$ and F1-scores on the training and test sets only show moderate improvements. This is because the distributions of the GOP scores of the training data will change after the model parameters are updated, and hence, the thresholds should be updated correspondingly. It is shown in Table II the MFC objective function increases consistently and significantly from 0.367 to 0.616 on the training set. Accordingly, the F1-score on the training set improves from 0.388 to 0.665, indicating that optimization of the model parameters according to MFC conforms to optimization of the empirical F1-score on the training set. For the test set, the F1-score rises from 0.381 to 0.488, suggesting that optimizing the empirical F1-score on the training set also leads to F1-score improvement on the unseen test set.

E. MFC Feature Space Training Results

The MFC-RDLC feature training results are listed in Table III. The training begins with a zero start of $\mathbf{B}$. Before iteratively optimizing $\mathbf{B}$. We also add ML adaptation of the baseline acoustic models to L2 training set as has been done in MFC model training. This helps to satisfy the assumption that the models are trained on correctly pronounced L2 data using ML estimation, whereby RDLC feature training is able to converge in fewer epochs. The HMM set contains 1592 Gaussians, thus the dimension of the posterior probability vector $\boldsymbol{\Phi}_t$ in (38) is 1592. When only the posterior probability vector of current frame $\boldsymbol{\Phi}_t$ is used, i.e., the context window

TABLE III
MFC FEATURE-SPACE TRAINING RESULTS

	Training		Test	
	$\mathcal{F}_{\mathrm{MFC}}$	F1	$\mathcal{F}_{\mathrm{MFC}}$	F1
MFC RDLC CXT=1	0.518	0.533	0.440	0.451
MFC RDLC CXT=3	0.557	0.574	0.446	0.453
MFC RDLC CXT=5	0.570	0.589	0.451	0.457
MFC RDLC CXT=7	0.587	0.598	0.462	0.462
MFC RDLC CXT=9	0.603	0.619	0.463	0.465
MFC RDLC CXT=AVG	0.605	0.622	0.471	0.474

TABLE IV
DETECTION RESULTS OF VARIOUS FEATURE/MODEL COMBINATIONS

Feature/Model	Training		Test	
	$\mathcal{F}_{\mathrm{MFC}}$	F1	$\mathcal{F}_{\mathrm{MFC}}$	F1
MFCC/ML	0.368	0.386	0.367	0.382
MFCC/ML(L2)	0.367	0.388	0.363	0.381
MFCC/MFC	0.616	0.665	0.454	0.488
RDLC/ML	0.605	0.622	0.471	0.474
RDLC/MFC	0.691	0.690	0.484	0.498

size is 1 ($\mathrm{CXT} = 1$ in Table III), F1-scores on the training and test sets are respectively 0.533 and 0.451, significantly better than those obtained by the baseline on normal MFCC features (0.386 and 0.382).

In speech recognition, using contextual acoustic information has shown its effectiveness in reducing recognition error rate. Here we expand the posterior probability vector $\mathbf{\Phi}_t$ in (38) by splicing the successive 3 posterior probability vectors $\mathbf{\Phi}_t$, $\mathbf{\Phi}_{t-1}$ and $\mathbf{\Phi}_{t+1}$ together (denoted as RDLC $\mathrm{CXT} = 3$ in Table III). F1-scores on the training and test sets are 0.574 and 0.453, respectively. By further increasing the context window size to $\mathrm{CXT} = 9$, F1-score on the test set (0.465) is 1.4% better than that of RDLC $\mathrm{CXT} = 1$, showing the effectiveness of context expansion in MFC feature training. However, considering that the MFC objective function (0.603) on the training set is much larger than that of RDLC $\mathrm{CXT} = 1$ (0.518), the improvement caused by such context expansion is fairly small.

We investigate the use of the splicing method in [27]: If the central (current) frame is at position 0, vectors are appended which are the average of the posterior vector at positions 1 and 2, at positions 3, 4 and 5, and at positions 6, 7, 8 and 9. The same is done to the left (positions -1 and -2, etc) so that the final vector is of size 1592×7 when there are 1592 Gaussians in the HMM set. As shown in Table III, this configuration (RDLC $\mathrm{CXT} = \mathrm{AVG}$) yields a better F1-score (about 1%) on the test set with a similar MFC objective improvement (0.605) on the training set compared with $\mathrm{CXT} = 9(0.603)$, indicating that this splicing method is more reliable and robust to overtraining. From the results of feature-space discriminative training, we conclude that optimizing the feature parameters according to MFC also conforms to optimization of F1-score on the training set and thereby the F1-score on the test set.

Finally we show the results of overall training in both the feature space and model space, i.e., performing MFC model update on MFC-RDLC features. We summarize mispronunciation detection results using ML or MFC training on raw MFCCs or RDLC features in Table IV. Note that the results of ML trained acoustic models on raw MFCC features is the baseline. The overall training result is about 2.4% better than feature-space training alone, 1.0% better than model-space training alone, and 11.6% better than the baseline.

F. Detection Results in Terms of Related Evaluation Metrics

In mispronunciation detection, performance evaluation metrics can be diversified. In this section we demonstrate detection results in terms of some other commonly used performance measures.

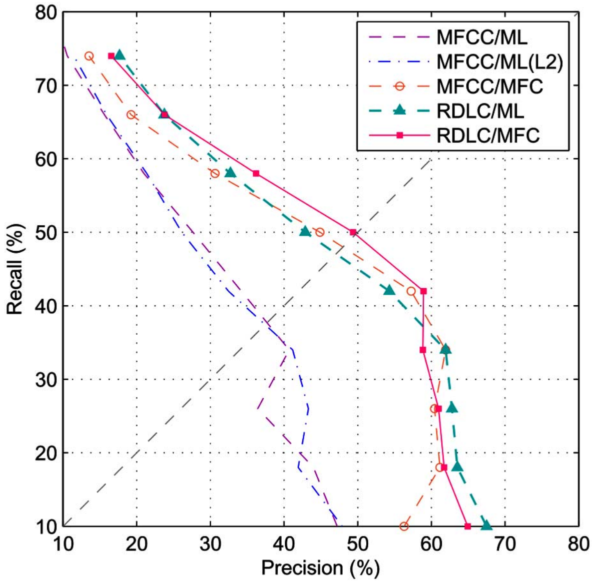

Fig. 3. ROC curves on L2 test set with various feature/model setups.

TABLE V
PRECISION AND RECALL IN MFC TRAINING (%)

Feature/Model	Training		Test	
	Precision	Recall	Precision	Recall
MFCC/ML	41.8	35.9	37.6	38.9
MFCC/ML(L2)	43.2	35.3	38.2	38.1
MFCC/MFC	79.5	57.2	57.1	42.7
RDLC/ML	71.0	55.4	50.6	44.6
RDLC/MFC	80.1	60.6	55.1	45.5

Precision and Recall: F1-score is the harmonic mean of Precision and Recall. Table V demonstrates the results of Precision and Recall obtained by the baseline, MFC model training, feature training and overall training. Results suggest that maximization of F1-score yields improvements of both Precision and Recall over the baseline. Fig. 3 demonstrates ROC curve (Precision and Recall curve) using various feature or model setups. It can be observed that MFC training also achieves better performance at the operating point where Precision equals Recall.

True Acceptance, False Acceptance, True Rejection and False Rejection: The number of these four outcomes are normalized by the total number of phone realizations in the data set and displayed in Table VI. It can be observed these four metrics can also be overall improved by MFC optimized features and models.

TABLE VI
DETECTION RESULTS IN TERMS OF TRUE ACCEPTANCE, FALSE ACCEPTANCE, TRUE REJECTION AND FALSE REJECTION (%)

Feature/Model	Train				Test			
	TA	FA	TR	FR	TA	FA	TR	FR
MFCC/ML	95.3	2.01	1.12	1.56	95.8	1.58	1.00	1.66
MFCC/ML (L2)	95.4	2.03	1.11	1.46	95.8	1.60	0.98	1.59
MFCC/MFC	96.4	1.34	1.79	0.46	96.5	1.48	1.10	0.83
RDLC/ML	96.2	1.40	1.73	0.71	96.3	1.43	1.15	1.12
RDLC/MFC	96.4	1.23	1.90	0.47	96.5	1.40	1.17	0.96

TABLE VII
DETECTION ERROR RATE, LIKELIHOOD, CONDITIONAL LIKELIHOOD IN MFC TRAINING

Feature/Model	DER (%)		Likelihood		Conditional Likelihood	
	Train	Test	Train	Test	Train	Test
MFCC/ML	3.57	3.24	–77.3	–77.0	–2.15	–2.02
MFCC/ML(L2)	3.48	3.18	–74.0	–74.1	–1.25	–1.22
MFCC/MFC	1.80	2.30	–79.1	–78.9	–4.09	–3.86
RDLC/ML	2.11	2.55	–88.6	–88.6	–3.85	–3.58
RDLC/MFC	1.71	2.36	–92.0	–92.1	–4.38	–4.14

TABLE VIII
RECOGNITION PERFORMANCE OF DIFFERENT ACOUSTIC MODELS

Data	Criterion	SER (%)	PER (%)
L1	ML	34.1	35.8
	MMI	27.9	32.8
	MPE	25.4	32.1
L2	ML	57.4	64.9
	MMI	43.9	55.4
	MPE	41.9	53.4

TABLE IX
DETECTION PERFORMANCE OF DIFFERENT ACOUSTIC MODELS

Criterion	Data	F1	Precision (%)	Recall (%)
ML	L1 training	0.382	37.6	38.9
ML	L2 training	0.381	38.2	38.1
MMI	L1 training	0.387	38.2	39.3
MMI	L2 training	0.346	37.9	31.9
MPE	L1 training	0.409	39.5	42.3
MPE	L2 training	0.375	33.4	42.7
MFC	L2 training	0.488	57.1	42.7

Detection Error Rate: Automatic mispronunciation detection can be viewed as a binary classification task. A phone segment can be classified as either a correct or erroneous pronunciation by the system. The number of correctly classified phone segments was used as an evaluation metric in [40], denoted as scoring accuracy (SA). Table VII demonstrates detection error rate (DER), i.e., the number of segments that have different detection result between the machine and human evaluator normalized by the total number of phone segments N. It is obvious that this number satisfy $DER = 1 - SA$ and is equal to the normalized sum of FA and FR:

$$DER = \frac{1}{N}(N_{FA} + N_{FR}) \times 100\%. \tag{53}$$

It is seen in Table VII that DER has significantly decreased on both the training and test sets by various MFC training setups, indicating that the improvement of F1-score also leads to reduction in DER.

G. Results on How MFC training affects Likelihood and Conditional Likelihood of the Canonical Phones

Likelihood: Improvement of likelihood on the training set is commonly used to indicate whether ML training is running correctly in developing a speech recognition system. Table VII shows the average frame log likelihood of the canonical phones using various acoustic model and feature setups. The average frame log likelihood of the baseline acoustic model on L2 training set and test set are respectively -77.3 and -77.0. ML adaptation on L2 training data show likelihood improvements. Based on the adapted models, MFC model training, feature training and overall training have seen significant likelihood decrease on L2 data set, though substantial F1-score improvements have been obtained. These results indicate that explicit F1-score optimization does not necessarily leads to model likelihood improvement. We may conjecture that increasing likelihood on training data is inconsistent with the maximization of F1-score in mispronunciation detection.

Conditional Likelihood: Table VII also presents the average of log posterior probabilities (log conditional likelihoods) of the canonical phones. Increment of the conditional likelihood is equivalent to the improvement of empirical phone classification accuracy. It can be seen MFC feature, model and overall training have significantly decreased conditional likelihood, i.e., expected phone classification accuracy. This means that F1-score optimization in mispronunciation detection is inconsistent with the reduction in phone classification error rate. We may conjecture that conditional maximum likelihood criterion, which can be used as an objective function to obtain better phone classification accuracy, is also not an desirable objective function for F1-score maximization in mispronunciation detection.

H. Comparisons of Mispronunciation Detection Results using Various Training Methods

In speech recognition, the acoustic models are often initialized using ML estimation and tuned with discriminative training criteria such as MMI and MPE. In this section we explore the effects of different model initializations and training objective functions in mispronunciation detection. As we have two databases: L1 native speech database and L2 non-native speech database, we trained the acoustic models on either L1 or L2 training set and used the derived models to detect pronunciation errors on L2 test set. We carried out syllable and phone output speech recognition experiments to evaluate recognition performance of the acoustic models. Language model is removed from decoding process to obtain a good evaluation of the acoustic resolution. Recognition performance in terms of Syllable Error Rate (SER) and Phone Error Rate (PER) are shown in Table VIII. The mispronunciation detection results on L2 test set in terms of F1-score, Precision and Recall are depicted in Table IX.

ML: ML training aims at maximizing likelihood of the observations given the transcriptions. The acoustic models trained with ML on L1 training data is our baseline. ML training on L2 training set corresponds to the ML adapted models on L2 training set. As presented, adaptation of baseline acoustic models on L2 training data showed likelihood improvement but did not make any clear differences to mispronunciation detection results.

MMI and MPE: The MMI objective function seeks to maximize the posterior probability of the correct utterance:

$$\mathcal{F}_{\mathrm{MMI}} = \sum_{r=1}^{R} \log \frac{p(\mathcal{O}_r|s_r)^{\kappa} P(s_r)^{\kappa}}{\sum_s p(\mathcal{O}_r|s)^{\kappa} P(s)^{\kappa}} \tag{54}$$

where R is the total number of training utterances, $\mathcal{O}_r$ is the observation sequence in utterance r, s_r is the reference transcription. $P(s)$ is the language model for sentence s. MPE aims at minimizing expected number of phone errors in a given hypothesis lattice, or equivalently maximizing the following objective function [15]:

$$\mathcal{F}_{\mathrm{MPE}} = \sum_{r=1}^{R} \frac{\sum_{s\in S} p(\mathcal{O}_r|s)^{\kappa} P(s)^{\kappa} A(s, s_r)}{\sum_{s'\in S} p(\mathcal{O}_r|s')^{\kappa} P(s')^{\kappa}} \tag{55}$$

where $A(s, s_r)$ is the raw accuracy score of word hypothesis s given the reference transcription s_r. More details about MPE can be found in [15]. We conducted MMI/MPE acoustic model training on both L1 and L2 training set. Both MMI and MPE training reduce recognition error rate, as shown in Table VIII, which indicates the acoustic models have been well trained by MMI and MPE before they are used for mispronunciation detection. However, as shown in Table IX, only MPE trained models on L1 data show F1-score improvement (2.7%). This indicates that discriminative training that aims at reducing WER does not explicitly lead to F1-score improvement in automatic mispronunciation detection.

VI. Conclusion and Future Work

We have proposed a discriminative training criterion which aims at directly maximizing F1-score in automatic mispronunciation detection. The training criterion is evaluated by feature-space discriminative training and model-space discriminative training. For model-space training, objective function maximization is achieved by using EBW form-like GMM-HMM updating equations based on the weak-sense auxiliary function theory. For feature-space discriminative training, we have trained a matrix that projects posteriors of Gaussians as compensations to traditional spectral features according to the MFC objective function. Mispronunciation detection experiments have shown the methods are effective in increasing F1-scores on both the training set and the test set. Further F1-score improvement can be obtained by MFC model parameter training on the newly trained features. It is also shown MFC training results in overall improvement in Precision and Recall, as well as improved mispronunciation detection accuracy, false acceptance rate, false rejection rate, true acceptance rate and true rejection rate.

The GOP based mispronunciation detection method can be also viewed as a two-class classification method. The method uses GOP score as feature input and use a pre-set threshold as the back-end classifier. The proposed method in this paper can be regarded as a representation learning method: We obtain more discriminating GOP scores by optimizing the input transform and GMM-HMM parameters according to MFC. Mispronunciation detection using GOP score and related features as input and DNN or SVM based classifier has been investigated in [6], [12]. Evaluation of the uses of MFC optimized GOP with better back-end classifiers could be remained for the further work.

In this paper we use GMM-HMM based acoustic models to compute GOP scores. Recent attempts in applying DNN based acoustic models to ASR have shown better speech recognition results. In mispronunciation detection, using DNN based acoustic models to compute GOP scores has been proposed and has shown better mispronunciation detection results [12]. In these cases, DNN parameters are often initialized using generative pre-training and then discriminatively fine-tuned, normally under the cross entropy (CE) criterion. In ASR, applying sequence discriminative criteria such as MMI, MPE to DNN training has shown lower WER [41]. This suggests that using a task-oriented objective function could be helpful in obtaining better performance. As shown, model training in mispronunciation detection using training criteria popular in ASR does not explicitly improve F1-score partly because no human judgements are involved. How the human-annotation results can be incorporated when applying DNN to mispronunciation detection remains an interesting research topic. We think the MFC objective function might be a better fine-tuning target for DNN based acoustic models in mispronunciation detection and the validation should be remained for the future work.

Acknowledgment

The authors of this paper would like to give special thanks to iFlyTek for the contributions to the mispronunciation labeling and ROC evaluation tools.

References

[1] H. Franco, L. Neumeyer, M. Ramos, and H. Bratt, “Automatic detection of phone-level mispronunciation for language learning,” in *Proc. Eurospeech*, 1999, pp. 851–854.

[2] S. Witt and S. Young, “Phone-level pronunciation scoring and assessment for interactive language learning,” *Speech Commun.*, vol. 30, no. 2–3, pp. 95–108, 2000.

[3] J. Zhang, C. Huang, M. Chu, F. K. Soong, and W. Ye, “Generalized segment posterior probability for automatic Mandarin pronunciation evaluation,” in *Proc. ICASSP*, 2007, pp. 201–204.

[4] F. Zhang, C. Huang, F. Soong, M. Chu, and R. Wang, “Automatic mispronunciation detection for Mandarin,” in *Proc. ICASSP*, 2008, pp. 2077–2080.

[5] Y. Tsubota, T. Kawahara, and M. Dantsuji, “Practical use of English pronunciation system for Japanese students in the CALL classroom,” in *Proc. ICSLP*, 2004, pp. 1689–1692.

[6] S. Wei, G. Hu, Y. Hu, and R. Wang, “A new method for mispronunciation detection using support vector machine based on pronunciation space models,” *Speech Commun.*, vol. 51, pp. 896–905, 2009.

[7] J. Tepperman and S. Narayanan, “Using articulatory representations to detect segmental errors in nonnative pronunciation,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 1, pp. 8–22, Jan. 2009.

[8] X. Qian, H. Meng, and F. Soong, “The use of DBN-HMMs for mispronunciation detection and diagnosis in L2 English to support computer-aided pronunciation training,” in *Proc. Interspeech*, 2012, pp. 775–778.

[9] A. Lee, Y. Zhang, and J. Glass, “Mispronunciation detection via dynamic time warping on deep belief network-based posteriorgrams,” in *Proc. ICASSP*, 2013, pp. 8227–8231.

[10] W. Hu, Y. Qian, and F. Soong, "A new DNN-based high quality pronunciation evaluation for computer-aided language learning (CALL)," in *Proc. Interspeech*, 2013, pp. 1886–1890.

[11] W. Hu, Y. Qian, and F. Soong, "A DNN-based acoustic modeling of tonal language and its application to Mandarin pronunciation training," in *Proc. ICASSP*, 2013, pp. 3230–3234.

[12] W. Hu, Y. Qian, and F. K. Soong, "A new neural network based logistic regression classifier for improving mispronunciation detection of L2 language learners," in *Proc. ISCSLP*, 2014, pp. 245–249.

[13] B. Juang and S. Katagiri, "Discriminative learning for minimum error classification," *IEEE Trans. Signal Process.*, vol. 40, no. 1, pp. 3043–3054, Jan. 1992.

[14] L. Bahl, P. Brown, P. Souza, and R. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," in *Proc. ICASSP*, 1986, vol. 11, pp. 49–52.

[15] D. Povey and P. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. ICASSP*, 2002, pp. 105–108.

[16] D. Povey, "Discriminative training for large vocabulary speech recognition," Ph.D. dissertation, Cambridge Univ., Cambridge, U.K., 2004.

[17] A. Fujino, H. Isozaki, and J. Suzuki, "Multi-label text categorization with model combination based on F1-score maximization," in *Proc. IJCNLP*, 2008, pp. 823–828.

[18] K. Dembczynski, W. Waegeman, W. Cheng, and E. Hullermeier, "An exact algorithm for F-Measure maximization," in *Proc. NIPS*, 2011, pp. 223–230.

[19] N. Ye, K. Chai, W. Lee, and H. Chieu, "Optimizing F-measures: A tale of two approaches," in *Proc. ICML*, 2012.

[20] D. Luo, Y. Qiao, N. Minematsu, Y. Yamauchi, and K. Hirose, "Analysis and utilization of MLLR speaker adaptation technique for learners' pronunciation evaluation," in *Proc. Interspeech*, 2009, pp. 608–611.

[21] W. Lo, S. Zhang, and H. Meng, "Automatic derivation of phonological rules for mispronunciation detection in a computer-assisted pronunciation training system," in *Proc. Interspeech*, 2010, pp. 765–768.

[22] H. Huang, J. Wang, and H. Abudureyimu, "Maximum F1-score discriminative training for automatic mispronunciation detection in computer-assisted language learning," in *Proc. Interspeech*, 2012, pp. 815–818.

[23] R. Duda, P. Hart, and D. Stork*, Pattern Classification*, 2nd Ed. ed. New York, NY, USA: Wiley-Interscience, 2000.

[24] N. Kumar and A. Andreou, A generalization of linear discriminant analysis in maximum likelihood framework Johns Hopkins Univ., Tech. Rep. JHU-CLSP Tech. Rep. No. 16, Aug. 1996.

[25] R. Gopinath, "Maximum likelihood modeling with Gaussian distributions for classification. in," in *Proc. ICASSP*, 1998, vol. II, pp. 661–664.

[26] G. Saon, M. Padmanabhan, R. Gopinath, and S. Chen, "Maximum likelihood discriminant feature spaces," in *Proc. ICASSP*, 2000, vol. II, pp. 1129–1132.

[27] D. Povey, B. Kingsbury, and L. Mangu *et al.*, "fMPE: Discriminatively trained features for speech recognition," in *Proc. ICASSP*, 2005, pp. 961–964.

[28] B. Zhang, S. Matsoukas, and R. Schwartz, "Discriminatively trained region dependent feature transforms for speech recognition," in *Proc. ICASSP*, 2006, vol. I, pp. 313–315.

[29] L. Deng, A. Acero, M. Plumpe, and X. D. Huang, "Large-vocabulary speech recognition under adverse acoustic environments," in *Proc. ICSLP*, 2000, vol. III, pp. 806–809.

[30] L. Deng, A. Acero, L. Jiang, J. Droppo, and X. D. Huang, "High-performance robust speech recognition using stereo training data," in *Proc. ICASSP*, 2001, vol. I, pp. 301–304.

[31] J. Droppo and A. Acero, "Maximum mutual information SPLICE transform for seen and unseen conditions," in *Proc. Interspeech*, 2005, pp. 989–992.

[32] L. Deng, J. Wu, J. Droppo, and A. Acero, "Analysis and comparison of two speech feature extraction compensation algorithms," *IEEE Signal Process. Lett.*, vol. 12, no. 6, pp. 477–480, Jun. 2005.

[33] M. Mahajan, A. Gunawardana, and A. Acero, "Training algorithms for hidden conditional random fields," in *Proc. ICASSP*, 2006, vol. I, pp. 273–276.

[34] D. Povey, D. Kanevsky, and B. Kingsbury *et al.*, "Boosted MMI for model and feature-space discriminative training," in *Proc. ICASSP*, 2008, pp. 4057–4060.

[35] J. Nocedal and S. Wright*, Numerical Optimization*. New York, NY, USA: Springer, 1999.

[36] M. J. F. Gales, "Model-based techniques for noise robust speech recognition," Ph.D. dissertation, Cambridge Univ., Cambridge, U.K., 1995.

[37] S. Wei, H. Wang, Q. Liu, and R. Wang, "CDF-matching for automatic tone error detection in Mandarin CALL system," in *Proc. ICASSP*, 2007, pp. 205–208.

[38] J. Cheng, "Automatic tone assessment of non-native Mandarin speakers," in *Proc. Interspeech*, 2013, pp. 1299–1302.

[39] H. Meng, Y. Lo, L. Wang, and W. Lau, "Deriving salient learners' mispronunciations from cross-language phonological comparisons," in *Proc. ASRU*, 2007, pp. 437–442.

[40] D. Luo, X. Yang, and L. Wang, "Improvement of segmental mispronunciation detection with prior knowledge extracted from large L2 speech corpus," in *Proc. Interspeech*, 2011, pp. 1593–1596.

[41] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence discriminative training of deep neural networks," in *Proc. Interspeech*, 2013, pp. 2345–2349.

Hao Huang received the B.E. degree from Shanghai Jiao Tong University, Shanghai, China, in 1999, the M.E. degree from Xinjiang University, Urumqi, China, 2004, and the Ph.D. degree from Shanghai Jiao Tong University, in 2008, respectively. He is currently an Associate Professor with School of Information Science and Engineering, Xinjiang University. His current research interests include speech and language processing, and multi-media human-computer interaction.

Haihua Xu received the B.E. degree from Harbin Higher Institute of Investment, Harbin China, 1998, and M.E. Degree from Huazhong University of Science and Technology, Wuhan, China, 2004, and the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, 2010, respectively. He is currently a research scientist at Speech Team of Temasek Laboratories in Nanyang Technological University, Singapore. His present research topic includes query-by-example spoken term detection, text-based keyword search from audio stream, and semi-supervised training.

Xianhui Wang received the B.E. degree from Xinjiang University, Urumqi, China, 2002, and the M.S. and Ph.D. degree from Xi'an Jiao Tong University, Xi'an, China, in 2005 and 2011, respectively. He is currently an Associate Professor with School of Information Science and Engineering, Xinjiang University. His current research interests include natural language processing, and machine learning.

Wushour Silamu is currently a Professor with School of Information Science and Engineering, Xinjiang University. His research interest covers speech recognition, speech synthesis, and multilingual information processing.