OPTIMIZATION OF THE DET CURVE IN SPEAKER VERIFICATION

*L. Paola Garcia-Perera, Juan A. Nolazco-Flores**

Tecnologico de Monterrey, Campus Monterrey
Computer Science Department
Monterrey, NL, Mexico

Bhiksha Raj, Richard Stern

Carnegie Mellon University
Language Technolgy Institute
Pittsburgh, PA, USA

ABSTRACT

Speaker verification systems are, in essence, statistical pattern detectors which can trade off false rejections for false acceptances. Any operating point characterized by a specific tradeoff between false rejections and false acceptances may be chosen. Training paradigms in speaker verification systems however either learn the parameters of the classifier employed without actually considering this tradeoff, or optimize the parameters for a particular operating point exemplified by the ratio of positive and negative training instances supplied.

In this paper we investigate the optimization of training paradigms to explicitly consider the tradeoff between false rejections and false acceptances, by minimizing the area under the curve of the detection error tradeoff curve. To optimize the parameters, we explicitly minimize a mathematical characterization of the area under the detection error tradeoff curve, through generalized probabilistic descent. Experiments on the NIST 2008 database show that for clean signals the proposed optimization approach is at least as effective as conventional learning. On noisy data, verification performance obtained with the proposed approach is considerably better than that obtained with conventional learning methods.

***Index Terms*—** Speaker verification, minimum verification error, discriminative training, joint factor analysis, detection cost function, detection error tradeoff.

1. INTRODUCTION

In this study we focus on one of the branches of speaker recognition: speaker verification (SV). The main purpose of SV is to provide a reliable decision, accept or reject a speaker, given a claimed identity and a recording of a spoken phrase. The two types of error that can happen are *false acceptances* (FA) – incorrect decisions to accept a speaker who is not actually the target speaker, and *false rejections* (FR) – incorrect rejection of speakers. Ideally the probability of both types of error would be zero; in practice the two can be traded off: the probability of false acceptance can be reduced at the cost of increased false rejection. Thus, for any given system, the operating point can be manipulated to obtain a desired ratio between false acceptances and false rejections. The entire range of possible operating points is characterized by a "detection error tradeoff" or DET curve. Typical DET curves showing how the operating point may be varied to modify the tradeoff between FA and FR are shown in Figure 1. It is desirable for both FA and FR errors to be low at any operating point. In other words, the DET curve must be close to or on the FA/FR axes. The performance of SV systems is usually evaluated in terms of how far the DET curve is from the axes, as characterized by

*Thanks to Tecnologico de Monterrey for funding.

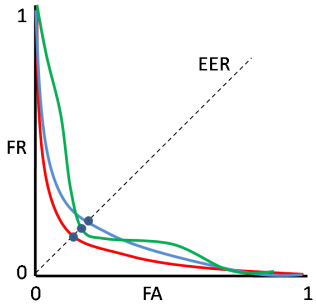

Fig. 1. DET curves for three different classifiers. Each point on the curve shows a different operating point specified by the FA and FR. The better classifier has the lower curve that is closer to the axes. The "Equal Error Rate" is the operating point shown by the intersection of the dotted diagonal line and the curve.

the *equal error rate* (EER) - the operating point at which the false acceptance and false rejection probabilities are equal [1], as this is assumed to present a balance between FA and FR. Intuitively, the lower the EER is, the closer the DET curve is to the axes. However this is not necessarily always true; the green curve in Figure 1 is clearly inferior to the blue one, yet has a better EER. A better measure of the overall performance of the system is the *area under the curve* (AUC) which measures the area between the axes and the DET curve.

There is generally a dichotomy between how SV systems are *trained* and how they are evaluated. Speaker verification is usually formulated as a likelihood-ratio test [2, 3, 4], computed using the estimated distribution of data from the target speaker and that from impostors. Traditional learning algorithms for SV systems such as maximum likelihood (ML) estimation [3, 4] and factor analysis [5, 6, 7] learn the parameters of these distributions to "fit" the training data, without explicitly considering the desired operating point of the system. *Discriminative* training paradigms do optimize classification performance, but usually only consider the specific operating point exemplified by the proportionp of "positive" (from the target speaker) and "negative" (not from the target speaker) training instances presented to them, which is but one point on the DET curve and may not be either the EER or the actual operating point eventually employed by the system.

In this paper we propose a discriminative training paradigm that *explicitly* learns the parameters of the distributions employed by the system to optimize the *entire* DET curve. The solution we propose is to minimize the AUC; this naturally also optimizes the performance at every operating point on the curve. To obtain an analytical so-

lution, we use the well known Wilcoxon Mann Whitney (WMW) statistic [8] as an equivalent to the AUC. The parameters of the distribution are then estimated using the (GPD) generalized gradient descent algorithm.

We show how this AUC approach can be incorporated into most current learning methods for SV systems including MVE (Minimum Verification Error) [9], and various flavors of factor analysis, including Joint-Factor Analysis (JFA) [6], i-vector representations [7], etc. We finally also present an experimental evaluation to demonstrate the method for MVE learning, where we find that the approach can result in comparable performance to conventional learning methods under clean conditions, but can result in dramatic improvements under noise.

The rest of the paper is as follows: in Section 2 we outline conventional training methods for speaker verification systems, in Section 3 we describe our approach to optimization of the DET curve, in Section 4 we describe our experiments and finally in Section 5 we present our conclusions.

2. TRAINING SPEAKER VERIFICATION SYSTEMS

The problem of speaker verification is to verify that a speaker S did indeed utter the speech signal χ that he claims to have spoken. The problem is typically treated as one of hypothesis testing. We define two hypotheses: a) the null hypothesis H_0 accepts the speaker as genuine and b) the alternative hypothesis H_1 rejects him.

The hypothesis testing is usually performed through a likelihood-ratio test. A parametric model with parameters Λ_S is defined for the distribution of data from S, and *counter* model with parameters $\Lambda_{\bar{S}}$ is specified for the class of impostors – the aggregate of all speakers who are not S. The difference in the log-likelihood of χ as given by the two models is compared to a threshold to choose the right hypothesis as shown in Equation 1.

$$\begin{aligned} \theta(\chi) &= \log\left(p\left(\chi;\Lambda_S\right)\right) - \log\left(p\left(\chi;\Lambda_{\bar{S}}\right)\right) \\ \text{if } \theta(\chi) > \tau: &\quad H_0 \\ \text{otherwise}: &\quad H_1 \end{aligned} \tag{1}$$

Two kinds of errors can happen in this process. An impostor's recording may incorrectly be verified as being from the target speaker, *i.e.* H_0 may be chosen, when the correct hypothesis is H_1. This is a *false acceptance* or FA. Alternately, a genuine recording by the target speaker may be rejected, *i.e.* H_1 is chosen where the correct hypothesis is H_0. This is a *false rejection* FA. As is clear from Equation 1, FA can be traded off for FR – by lowering the threshold τ, a larger number of recordings will trigger the null hypothesis H_0. While this increases the probability that genuine recordings by speaker will not be rejected, it also increases the probability that impostor recordings will be accepted.

This trade-off between FA and FR is usually characterized by a DET curve, such as the ones shown in Figure 1. In principle the DET curve must show the probability of FA against the probability of FR; in practice it is obtained empirically by estimating the FR and FA probabilities from large collections of genuine and impostor recordings. The DET curve characterizes the reliability of the system – a DET curve that lies closer to the axes signifies both lower FA and lower FR.

The problem of *training* a reliable SV system is then transformed to learning the parameters Λ_S and $\Lambda_{\bar{S}}$ of an appropriately chosen parametric form of the probability distributions.

We first note that each recording χ actually comprises a *sequence* of feature vectors, typically mel-frequency cepstral vectors augmented by their velocity and acceleration. Thus $\chi = X_1, X_2, \cdots, X_T$, where X_i is the i^{th} vector in the sequence.

The most common form of probability distribution for χ treats the individual X_i vectors as IID with a probability distribution given a Gaussian mixture model, *i.e.*

$$P(\chi;\Lambda_C) = \prod_i \sum_k w_k^C \mathcal{N}(X_i;\mu_k^C,\Sigma^C,k),$$

where C is either S or $\bar{S}$, and w_k^C, μ_k^C and Σ_k^C are the mixture weight, mean and covariance (usually assumed to be a diagonal matrix) of the k^{th} Gaussian in the mixture. Thus $\Lambda_C = \{w_k^C, \mu_k^C, \Sigma_k^C \ \ \forall k\}$. Training the SV system is hence equivalent to learning GMM parameters for both S and $\bar{S}$.

Learning GMM parameters can generally be performed by the expectation maximization algorithm, given a suitably large amount of training data. For SV systems, however, while an arbitrarily large amount of training data may be available to learn the counter model $\Lambda_{\bar{S}}$, the training data available to learn the model Λ_S for any speaker are usually limited. The common solution therefore is to learn a robust counter model $\Lambda_{\bar{S}}$, which in this context is usually referred to as a Universal Background Model (UBM), from a large collection of speech from a large number of speakers, and to *adapt* the UBM to the training data from the speaker to obtain Λ_S.

In the ideal case where the training data from the speaker are recorded over the same kinds of channels that will be employed to record test data that must be verified, *maximum a posteriori* (MAP) adaptation assuming conventional conjugate priors for the mixture weights, means and variances has been found to be a good solution for estimating Λ_S [10]. However, for non-ideal cases, the state of the art solution utilizes *joint factor analysis* (JFA) [5, 6] to provide the priors for the parameters. The purpose of JFA is to separate out *speaker* and *channel* factors that contribute to the distribution. Channel factors, which do not relate to the identity of the speaker, can be marginalized out during the computation of likelihoods for the likelihood ratio test.

In the JFA framework, the means of all the Gaussians in a GMM, Λ are concatenated into a single vector, termed a *supervector* $M = [\mu_1 \| \mu_2 \| \cdots]$ (here we have not explicitly shown the superscript representing the speaker for generality). The supervector $M_{S,H}$ representing the GMM for the distribution of data over *each channel type* H by a speaker S is assumed to be composed from a collection of factors as:

$$M_{S,H} = m + Vy_S + Ux_{S,H} + Dz_S, \tag{2}$$

m is a global mean across all speakers, V is a set of *eigenvoice* loadings representing the subspace over which speaker-specific components of $M_{S,H}$ lie, and U is a set of *eigenchannel* loadings representing the subspace over which the channel specific components of $M_{S,H}$ lie. D is a diagonal matrix. y_S is a normally distributed vector (with 0 mean and unit variance) representing speaker factors specific to speaker S, and $x_{S,H}$ is a normally distributed vector representing channel factors specific to recordings of speaker S over channel H. z is also a normally distributed vector representing residual error. The Baum-Welch statistics for learning the loadings V, U and D, the EM procedures for learning the factors y_S, x_H, z_S, as well as the methods for classification with the resulting model in a manner that factors out the interfering contributions of the channel are well laid out in [6, 5].

3. LEARNING PARAMETERS TO MINIMIZE THE AUC OF THE DET CURVE

We note that the learning methods outlined in Section 2 train the distributions of the individual classes S and $\bar{S}$ without explicitly considering the fact that the actual task is one of discrimination. Needless to say, discriminative versions of these learning algorithms also exist.

Discriminative [11] and maximum margin [12] methods for both learning and adaptation of GMM parameters have been proposed in the literature. Similarly, discriminative versions of JFA and its variants have also been proposed, *e.g.* [13]. All of these methods attempt to learn model parameters to maximize the empirical classification error on the training data provided. Consequently, they naturally optimize the performance at a *specific* operating point – that characterized by the relative (possibly weighted) proportions of positive and negative examples provided to them. This operating point may not be related at all to the actual operating point at which the system is actually eventually evaluated. Moreover, we would ideally not like to *commit* to an operating point – we would like to optimize the performance at *every* operating point. In other words, we do not merely wish to lower the DET curve along a specific diagonal line from origin (such as the EER line shown in Figure 1; we would like to explicitly lower the *entire* DET curve. The way we will do this is by explicitly minimizing the AUC – the area under the DET curve.

Consider a binary classifier that attempts to classify as data as belonging to a class C or not. Let $\mathcal{H}$ and $\mathcal{W}$ be two sets of data belonging respectively to C and $\bar{C}$ (*i.e.* not in C). The empirical AUC of a classifier that computes a score $\theta(\chi)$ to determine if any data instance χ belongs to C is given by Equation 3.

$$G(\Lambda) = 1.0 - \frac{\sum_{\chi\in\mathcal{H}}\sum_{\hat{\chi}\in\mathcal{W}} 1(\theta(\chi) > \theta(\hat{\chi}))}{|\mathcal{H}||\mathcal{W}|} \tag{3}$$

where 1 is the indicator function. The term being subtracted from 1.0 in Equation 3 is the well-known Wilcoxon-Mann-Whitney statistic [8]. In the equation above we have written $G(\Lambda)$ to be a function of $\Lambda = \Lambda_C \cup \Lambda_{\bar{C}}$, where Λ_C and $\Lambda_{\bar{C}}$ are the parameters of any models associated with C and $\bar{C}$, to aid us in the explanation below.

We can minimize the AUC by minimizing the function given in Equation 3. Equation 3, computed over the training set, is thus a new objective function to be minimized. The AUC function specified by Equation 3 is not smooth however, since it is the sum of discontinuous indicator functions. In order to optimize it, a smooth, differentiable version of this objective function is needed.

To do so, we replace the indicator functions $1(a > b)$ by a sigmoid function, following an approach that is commonly used in discriminative training methods, *e.g.* [14], as in Equation 4,

$$R(a,b) = \frac{1}{1 + \exp\left(-\gamma\ \ \varphi(a,b)\right)}, \tag{4}$$

where γ governs the steepness of the sigmoid and controls the learning rate and φ is the distance $\varphi(a,b) = a - b$.

Thus, the modified AUC function to optimize is:

$$G(\Lambda) = 1.0 - \frac{\sum_{\chi\in\mathcal{H}}\sum_{\hat{\chi}\in\mathcal{W}} R(\theta(\chi),\theta(\hat{\chi}))}{|\mathcal{H}||\mathcal{W}|} \tag{5}$$

The modified AUC function of Equation 5 must be appropriately customized to the type of model being considered. It can then be optimized using the generalized probabilistic descendent (GPD) algorithm. Let $\mathbf{X}$ represent the complete set of training instances: $\mathbf{X} = \mathcal{H} \cup \mathcal{W}$. The GPD updates are performed according to the following:

$$\Lambda_{t+1} = \Lambda_t - \epsilon\nabla L(\mathbf{X},\Lambda) \tag{6}$$

$$\nabla L(\mathbf{X},\Lambda) = -\frac{1}{|\mathcal{H}||\mathcal{W}|}\sum_{\chi\in\mathcal{H}}\sum_{\hat{\chi}\in\mathcal{W}}\gamma(1-R)\left[\frac{\partial\theta(\chi)}{\partial\Lambda} - \frac{\partial\theta(\hat{\chi})}{\partial\Lambda}\right] \tag{7}$$

In the above equation R is a short-hand notation for $R(\theta(\chi),\theta(\hat{\chi}))$. ϵ is a learning rate parameter.

The above formalism can generally be used in all formulations of speaker verification with appropriate customization of the objective function. Below we consider two approaches as an illustration.

3.1. Minimum Verification Error

In the minimum-verification error approach, GMM parameters for individual speakers are optimized to minimize empirical verification error on the training set [15]. Replacing the objective function usually used in MVE training, we get the modified objective function:

$$G(\Lambda) = 1.0 - \frac{\sum_{\chi\in\mathcal{H}}\sum_{X\in\chi}\sum_{\hat{\chi}\in\mathcal{W}}\sum_{\hat{X}\in\hat{\chi}} R(\theta(X),\theta(\hat{X}))}{\sum_{\chi\in\mathcal{H}} L_\chi \sum_{\hat{\chi}\in\mathcal{W}} L_{\hat{\chi}}} \tag{8}$$

where L_χ is the number of feature vectors in χ. Note that the original AUC formulation of Equation 3 only considers misclassifications – both FA and FR, in keeping with the conventional formulation of MVE. The "soft" version of the AUC given by Equation 8 also naturally conforms to this formulation.

Using the representation $\sum_{\chi\in\mathcal{H}} L_\chi = |\mathcal{H}|$, $\sum_{\chi\in\mathcal{W}} L_\chi = |\mathcal{W}|$, the GPD update rule for any parameter ϕ of the distributions is now given by $\phi_{t+1} = \phi_t - \epsilon\nabla_\phi L(\mathbf{X},\Lambda)$, where

$$\nabla_\phi L(\mathbf{X},\Lambda) = -\frac{1}{|\mathcal{H}||\mathcal{W}|} \sum_{\chi\in\mathcal{H}}\sum_{X\in\chi}\sum_{\hat{\chi}\in\mathcal{W}}\sum_{\hat{X}\in\hat{\chi}}\gamma(1-R)\nabla_\phi l(X,\hat{X},\Lambda) \tag{9}$$

and $\nabla_\phi l(X,\hat{X},\Lambda)$ is a local gradient with respect to ϕ at $\chi,\hat{\chi}$ and has the form given by Equation 10.

$$\nabla_\phi l(X,\hat{X},\Lambda) = -\frac{\partial\theta(X)}{\partial\phi} + \frac{\partial\theta(\hat{X})}{\partial\phi} \tag{10}$$

$\frac{\partial\theta(X)}{\partial\phi}$ represents the derivative of the log-likelihood-difference given by the Gaussian mixture models for the target speaker and the universal background model for vector X with respect to ϕ. The update rules for the individual parameters w_k^S, μ_k^S and Σ_k^S are obtained by plugging in $\frac{\partial\theta(X)}{\partial w_k^S}$, $\frac{\partial\theta(X)}{\partial\mu_k^S}$ and $\frac{\partial\theta(X)}{\partial\Sigma_k^S}$ respectively into Equation 10. These are relatively straightforward to derive, and we omit the details for brevity. Details may be found in the appendix of the extended version of this submission at [16].

3.2. JFA

In the case of JFA the set of parameters to be learned are of two kinds. The *global* parameters include V, the speaker loadings, U, the channel loadings, and D, the diagonal error scaling matrix. The *specific* parameters include y_S, which is specific to a speaker S and $x_{S,H}$, which is specific to the speaker-channel combination S, H.

Hence, two distinct learning problems must be addressed: learning the global loadings from a large collection of speaker recordings over a variety of channels, and learning specific factors for individual speakers. The global parameters must learn the overall characteristics of the the speaker and channel subspaces. Specific parameters must be learned to customize a model to a specific speaker given training data for the speaker.

The AUC objective function must be appropriately customized in each case. In all cases, the discriminant function $\theta()$ in Equation 1 is specified as in Equation 11.

$$\begin{aligned}\theta(\chi) = &\log \ \ p(\chi; V, U, D, y_S(\chi), x_{H(\chi),S(\chi)}) \\ &- \log \ \ p(\chi; \lambda_{\bar{S}}, U, x_{H(\chi),S(\chi)})\end{aligned} \tag{11}$$

Here $S(\chi)$ represents the speaker S represented in the recording χ. $H(\chi)$ represents the recording channel in χ. The equation above explicitly indicates that the log-likelihood for the model of speaker S is computed using Gaussian parameters from the supervector $M = m + Vy_{S(\chi)} + Ux_{S(\chi),H(\chi)} + Dz$, whereas the parameters of the "counter" model for any recording are obtained from $M' = m + Ux_{S(\chi),H(\chi)} + Dz$, which only considers the universal mean m adjusted by the channel factors which customize them to the recording χ. The global mean m is derived from the universal background model $\lambda_{\bar{S}}$.

3.2.1. Global parameters: Estimating the loadings

Let $\mathbf{X}$ represent a large collection of recordings χ obtained from a large number of speakers. Let $\mathcal{S}$ be the set of all speakers represented in $\mathbf{X}$. Let $\mathbf{X}_S$ represent the subset of $\mathbf{X}$ representing recordings from speaker S and $\mathbf{X}_{\bar{S}}$ be recordings from remaining speakers, *i.e.* $\mathbf{X} = \mathbf{X}_S \cup \mathbf{X}_{\bar{S}}$. $\mathbf{X}$ can be partitioned in this manner in as many ways as there are speakers in $\mathcal{S}$.

To learn global parameters, we define the AUC objective function as given in Equation 12.

$$\begin{aligned}WMW(\Lambda) &= \sum_{S \in \mathcal{S}} \frac{\sum_{\chi \in \mathbf{X}_S} \sum_{\chi \in \mathbf{X}_{\bar{S}}} R(\theta_S(\chi), \theta_S(\hat{\chi}))}{|\mathbf{X}_S||\mathbf{X}_{\bar{S}}|} \\ G(\Lambda) &= 1.0 - WMW(\Lambda)\end{aligned} \tag{12}$$

As before, the GPD update rule for any global parameter ϕ is given by $\phi_{t+1} = \phi_t - \epsilon \nabla_\phi L(\mathbf{X}, \Lambda)$, where

$$\nabla_\phi L(\mathbf{X}, \Lambda) = - \sum_{S \in \mathcal{S}} \frac{\sum_{\chi \in \mathbf{X}_S} \sum_{\chi \in \mathbf{X}_{\bar{S}}} \gamma(1 - R) \nabla_\phi l(\chi, \hat{\chi}, \Lambda)}{|\mathbf{X}_S||\mathbf{X}_{\bar{S}}|} \tag{13}$$

and $\nabla_\phi l(\chi, \hat{\chi}, \Lambda)$ is a local gradient with respect to ϕ at $\chi, \hat{\chi}$.

$$\nabla_\phi l(\chi, \hat{\chi}, \Lambda) = -\frac{\partial \theta(\chi)}{\partial \phi} + \frac{\partial \theta(\hat{\chi})}{\partial \phi} \tag{14}$$

To derive the update rules for individual parameters, it is sufficient to obtain the derivatives $\frac{\partial \theta(\chi)}{\partial \phi}$ with respect to the corresponding parameters. The case was studied in [17] and it is extended in the equations given below.

Consider $P = D^2 + VV^\top$ and $Q = D^2 + UU^\top$. Then,

$$\frac{\partial \theta(X)}{\partial m} = -[P^{-1}(\chi - m)] \tag{15}$$

$$\frac{\partial \theta(X)}{\partial V} = [(P^{-1}V] - [P^{-1}(\chi - m)][(\chi - m)^T P^{-1} V] \tag{16}$$

$$\frac{\partial \theta(X)}{\partial U} = [Q^{-1}U] - [Q^{-1}(\chi - m)][(\chi - m)^T Q^{-1} U] \tag{17}$$

$$\frac{\partial \theta(X)}{\partial D} = \frac{1}{2} \left\{ [P^{-1}] - [P^{-1}(\chi - m)]^2 \right\} \tag{18}$$

3.2.2. Estimating Specific Parameters

To learn the specific parameters for a particular speaker S, the AUC objective is defined simply as

$$G(\Lambda_S) = 1.0 - \frac{\sum_{\chi \in \mathbf{X}_S} \sum_{\chi \in \mathbf{X}_{\bar{S}}} R(\theta_S(\chi), \theta_S(\hat{\chi}))}{|\mathbf{X}_S||\mathbf{X}_{\bar{S}}|} \tag{19}$$

Note that unlike Equation 12 which includes an outer summation over all speakers, Equation 19 only considers a single speaker S.

Once again, the GPD update rule for any global parameter ϕ is given by $\phi_{t+1} = \phi_t - \epsilon \nabla_\phi L(\mathbf{X}, \Lambda)$. As before, the GPD update rule for any global parameter ϕ is given by $\phi_{t+1} = \phi_t - \epsilon \nabla_\phi L(\mathbf{X}, \Lambda)$, where

$$\nabla_\phi L(\mathbf{X}, \Lambda) = - \frac{\sum_{\chi \in \mathbf{X}_S} \sum_{\chi \in \mathbf{X}_{\bar{S}}} \gamma(1 - R) \nabla_\phi l(\chi, \hat{\chi}, \Lambda)}{|\mathbf{X}_S||\mathbf{X}_{\bar{S}}|} \, .$$

where $\nabla_\phi l(\chi, \hat{\chi}, \Lambda)$ is defined as in Equation 14.

To derive the update rules for individual parameters y_S, $x_{S,H}$ and z_S, it is sufficient to obtain the derivatives $\frac{\partial \theta(\chi)}{\partial y_S}$, $\frac{\partial \theta(\chi)}{\partial x_{S,H}}$ and $\frac{\partial \theta(\chi)}{\partial z_S}$, and employ these in the GPD update rules. Although these derivatives are straight forward to obtain, we omit the details of the equations here for brevity. They may be found in the appendix of the extended version of this submission at [16]. In practice the speaker and channel factors y_S, $x_{H,S}$ and z_S can also be estimated using conventional EM estimate rules.

Note that the estimation of global parameters also requires estimation of specific parameters, since the update rules for the former require the latter. Thus, estimation of global parameters involves estimation of both global and specific parameters for all the speakers in the training set. To learn a model for a new speaker for whom a small amount of training data have been made available, only the specific parameters need be learned employing the already-known global parameters.

Although we have explained the above in terms of JFA, the same formulation can also be used to learn i-vector representations [7], which are essentially the same as the above without explicit separation of channel factors. AUC-optimized PLDA [18] based representations can be derived similarly to the rules given above, with the modification that the GPD rules will now employ partial derivatives with respect to PLDA parameters.

4. EXPERIMENTS AND RESULTS

We ran experiments to evaluate the proposed AUC-minimization approach. Two experiments were run. In the first we compared the performance of conventional MAP and JFA based learning with JFA optimized using the AUC criterion on speech recordings, where the noise conditions in the training and test data were matched. In the second we compare the performance of AUC-minimization base MVE against conventional methods on mismatched conditions, where the test data are noisy.

4.1. Experimental Setup

We employed the NIST Speaker Evaluation 2004 and 2008 database [19] to complete this study. Moreover, we followed the evaluation rules (for instance, neither choosing other target model as anti-model, nor using other target data to estimate the current target model). For the feature extraction, a short-time 256-pt Fourier analysis is performed on a 25ms analysis window and 10ms frame rate. The feature vector (token) consists of 49 Mel Frequency Cepstral Coefficients (MFCCs), including delta and double delta coefficients. We included a frame removal criterion that encloses the concept of eliminating the low energy frames that do not provide information about the identity of the person.

4.2. Baseline framework

We first employ a baseline result using both MAP and JFA to model each speaker [20]. We first compute a gender-dependent and target-independent "anti-model", a UBM trained from a pool of raw speech (NIST Speaker Evaluation 2004 core database). This model captures the characteristics of all the data vectors of the users not belonging to the target set of speakers to be evaluated. The *expectation maximization* (EM) algorithm is used to estimate the GMM parameters of the anti-model. For the MAP-based baseline, the models for target speakers in the evaluation set are obtained by MAP adaptation of the UBM. For the JFA baseline, the speaker and channel factors were learned from the pool of impostors using EM and adapted to individual speakers by estimation of factors. The code for JFA was obtained from the implementation of the Speech Processing Group at the Brno University of Technology [21] and used in part or whole by [22, 23, 18] well known sites. All verification tests were performed under the hypothesis test framework.

4.3. Experiments on clean speech

In the first recording all training and test recordings were noise-free, although collected over varied channels. This is the standard setup for the NIST 2008 test We compared the baseline classification performance to that obtained with AUC-minimized JFA. For the AUC-optimized JFA, the loading matrices V, U and D were computed to minimize the AUC. The factors y_S and $x_{H,S}$ were learned conventional. For the test speakers, target users were enrolled conventionally, through MAP adaptation and computation of the factors: x_S, $y_{H,S}$ and z.

For the AUC-optimized training, we need: a) target and impostor tokens, b) an initial target model and the anti-model. The initial target model was provided by the adapted models in the baseline (MAP). One of the difficulties of discriminative training is the increase of computing time compared to generative modeling (ML approach). In the ideal case, all impostor tokens used to train the "counter-model" should be used to update the models. However, this problem is infeasible. To solve this issue, we selected just a chunk of the tokens available in the database for the AUC-optimized training. In this case, following NIST evaluation constraints of not allowing the use of other target speakers within the same database to model target speakers, NIST Speaker Evaluation 2004 core database was used instead to obtain the counter model tokens.

Table 1 shows the results obtained. The results are consistent with comparisons performed by other researchers: JFA outperforms both baseline MAP learning as well MVE learning significantly. We note that the JFA uses an optimized implementation by [21] and the results too have been optimized empirically and may be considered to be competitive for the particular data setup used. All performance numbers are noted to improve as the amount of training data used to learn the base UBM increases.

More importantly, we note that the AUC-optimized classifier actually outperforms conventionally trained models both for MVE and JFA. In fact, over multiple runs of the experiment with different initializations and parameterizations, the trend of results in Table 1 were maintained.

Baseline MAP	15.95
Baseline JFA	12.07
MVE	13.51
AUC-MVE	13.21
AUC-JFA	11.93

Table 1. EER for MVE and JFA

4.4. Experiments on noisy speech

In the second experiment we compared baseline techniques to AUC-minimized training on noisy speech. Experiments were performed using speech corrupted to a variety of SNRs (10dB, 15dB, and a cocktail of 0-15dB), all of them using babble noise.

For all experiments, we used 100 male registered users. Following NIST 2008 Evaluation rules, the probability of being a target, P_{target}, is 0.01 and the probability of being a impostor, $P_{impostor}$, is 0.99. Moreover, this study presents conclusive experiments for a 512 component GMM in all cases. No normalization was used after the score computation to observe the full capabilities of MVE and JFA approach.

Table 2 compares JFA to AUC-optimized MVE on noisy speech of various SNRs. We observe that AUC-optimized MVE learning consistently outperforms both regular MVE training and JFA on noisy speech. The results are consistent across all noise conditions. Table 3 compares AUC-optimized JFA and MVE with conventional JFA on speech at 10dB SNR. Again, we find that the AUC-optimized algorithms outperform conventional training. Interestingly, AUC-MVE outperforms AUC-JFA in this case.

System	**10dB**	**15dB**	**cocktail 0-15dB**
MAP Baseline	18.01	-	-
JFA	17.23	16.79	27.3
MVE-AUC	16.12	16.02	23.5

Table 2. EER of the noisy task (babble noise).

Table 3 shows an example of noisy speech including JFA-AUC. Once again, we observe that AUC-optimized MVE learning outperforms MVE and jFA on noisy conditions.

MAP	JFA	JFA-AUC	MVE-AUC
18.01	17.23	16.92	16.12

Table 3. EER on 10dB speech (babble noise).

5. DISCUSSION AND CONCLUSION

The experiments in the previous section are by no means exhaustive; however they are clearly indicative. AUC-based optimization is competitive with conventional training methods in the worst case and can outperform them greatly under other circumstances. In the previous experiment, we have not evaluated AUC-optimized JFA on

noisy speech, primarily for lack of computational resources; we believe that the results obtained with that procedure may have been better still. Moreover, the results reported above are only EER results. AUC optimization actually optimizes performance at all operating points. Presumably, if one were to consider the combined results at all operating points, the benefits of AUC-based learning would be even greater. The AUC-optimized learning techniques employed above are not optimized; only some of the parameters that could be optimized in this manner have been optimized. We are in the process of investigating and optimization of learning for other parameters and expect these will provide additional improvements.

As noted earlier, AUC-optimization can also be employed for other learning and modeling techniques, *e.g.* i-vector based representations [7] and PLDA [24, 25]. Generalizations can also consider multi-class classification results such as those employed for speaker *identification*. These too are current areas of research.

Perhaps another lesson to be derived from our experiments is that direct optimization of an objective function that directly relates to the actual task being performed. This is not an epiphany – this fact has always been known; all we provide is some confirmation.

Finally, if the only objective is the performance at a specific operating point (as specified by the ratio of FA and FR), then the AUC objective function could be modified to consider only that operating point. It is to be expected that the performance at that operating point will improve, possibly at the cost of performance at other operating points. This too remains an area of investigation.

6. REFERENCES

[1] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The det curve in assessment of detection task performance," Tech. Rep., DTIC Document, 1997.

[2] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[3] F. Bimbot, J. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, and DD Petrovska-Delacretaz, "Reynolds (2004) A tutorial on text-independent speaker verification," *EURASIP Journal on Applied Signal Processing*, vol. 4, pp. 430–451.

[4] D. Petrovska-Delacrétaz, A. El Hannani, and G. Chollet, "Text-independent speaker verification: state of the art and challenges," *Progress in nonlinear speech processing*, pp. 135–169, 2007.

[5] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1435–1447, 2007.

[6] P. Kenny, P. Oueleet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of inter-speaker variability in speaker verification," *IEEE Trans. ASLP*, vol. 16, pp. 980–988, 2008.

[7] Najim Dehak, Patrick J. Kenny, Rda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, 2010.

[8] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *Annals of Mathematical Statistics*, vol. 18:1, pp. 50–60, 1947.

[9] CH Lee, "A unified statistical hypothesis testing approach to speaker verification and verbal information verification," in *Proc. COST,Workshop on Speech Technology in the Public Telephone Network: Where are we today?*, Greece, September 1997, vol. 250, pp. 63–72.

[10] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of markov chains," *IEEE Trans. on Speech and Audio Processing*, vol. 2, pp. 291–299, Apr. 1994.

[11] D. Povey, PC Woodland, and MJF Gales, "Discriminative map for acoustic model adaptation," in *IEEE Intl. Conf. on Acoustics, Speech, and Sig. Proc. (ICASSP)*, 2003, vol. 1, pp. I–312.

[12] F. Sha and L.K. Saul, "Large margin gaussian mixture modeling for phonetic classification and recognition," in *IEEE Intl. Conf. on Acoustics, Speech and Sig. Proc. (ICASSP)*, 2006.

[13] L. Burget, O. Plchot, S. Cumani, O. Glembek, P. Matejka, and N. Brummer, "Discriminatively trained probabilistic linear discriminant analysis for speaker verification," 2012, Proceedings of ICASSP.

[14] B.-H. Juang, W. Chou, and C.-H. Lee, "Minimum classification error rate methods for speech recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 5, pp. 257–265, May 1997.

[15] A. E. Rosenberg, "Speaker verification using minimum verification error training," 1998, Proceedings of ICASSP.

[16] "Optimization of the det curve in speaker verification(extended version)," `http://mlsp.cs.cmu.edu/publications/pdfs/slt2012AUCSV.pdf`.

[17] L.K. Saul and M.G. Rahim, "Maximum likelihood and minimum classification error factor analysis for automatic speech recognition," *Speech and Audio Processing, IEEE Transactions on*, vol. 8, no. 2, pp. 115–125, 2000.

[18] L. Burget, O. Plchot, S. Cumani, O. Glembek, P. Matejka, and N. Brummer, "Discriminatively trained probabilistic linear discriminant analysis for speaker verification," in *IEEE Intl. Conf. on Acoustics, Speech and Sig. Proc. (ICASSP)*, 2011.

[19] A.F. Martin and C.S. Greenberg, "NIST 2008 Speaker Recognition Evaluation: Performance Across Telephone and Room Microphone Channels," in *Proc. Interspeech*, 2009.

[20] D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication*, vol. 17, no. 1-2, pp. 91–108, 1995.

[21] L. Burget, M. Fapso, and V. Hubeika, "BUT system for NIST 2008 speaker recognition evaluation," in *Interspeech*, 2009.

[22] N. Scheffer, L. Ferrer, M. Graciarena, S. Kajarekar, E. Shriberg, and A. Stolcke, "The SRI NIST 2010 speaker recognition evaluation system," in *IEEE Intl. Conf. on Acoustics, Speech and Sig. Proc. (ICASSP)*, 2011, pp. 5292–5295.

[23] O. Glembek, L. Burget, N. Dehak, N. Brummer, and P. Kenny, "Comparison of scoring methods used in speaker recognition with joint factor analysis," in *IEEE Intl. Conf. on Acoustics, Speech and Sig. Proc. (ICASSP)*, 2009.

[24] P. Matejka, O. Glembek, F. Castaldo, MJ Alam, O. Plchot, P. Kenny, L. Burget, and J. Cernocky, "Full-covariance UBM and heavy-tailed PLDA in i-vector speaker verification," in *IEEE Intl. Conf. on Acoustics, Speech and Sig. Proc. (ICASSP)*, 2011.

[25] P Kenny, "Bayesian Speaker Verification with Heavy-Tailed Priors," *keynote presentation, Odyssey Speaker and Language Recognition Workshop Brno, Czech Republic*, 2010.