SPEAKER VERIFICATION USING MINIMUM VERIFICATION ERROR TRAINING

Aaron E. Rosenberg *Olivier Siohan* *S. Parthasarathy*

AT&T Labs – Research
180 Park Avenue, Florham Park
NJ 07932-0971, USA

ABSTRACT

We propose a Minimum Verification Error (MVE) training scenario to design and adapt an HMM-based speaker verification system. By using the discriminative training paradigm, we show that customer and background models can be jointly estimated so that the expected number of verification errors (false accept and false reject) on the training corpus are minimized. An experimental evaluation of a fixed password speaker verification task over the telephone network was carried out. The evaluation shows that MVE training/adaptation performs as well as MLE training and MAP adaptation when performance is measured by average individual equal error rate (based on *a posteriori* threshold assignment). After model adaptation, both approaches lead to an individual equal error-rate close to 0.6%. However, experiments performed with *a priori* dynamic threshold assignment show that MVE adapted models exhibit false rejection and false acceptance rates 45% lower than the MAP adapted models, and therefore lead to the design of a more robust system for practical applications.

1. INTRODUCTION

A speaker verification task is essentially a hypothesis testing problem where, given some speech observations, it must be decided whether the claimed identity of the speaker is correct. The speaker characteristics are usually represented using the probability density function (pdf) of some speech related observation vectors. The hypothesis testing problem therefore consists of deciding between two hypotheses regarding some parameters λ of the probability density function. The two hypotheses are the *null* hypothesis, denoted by $H_0 : \lambda = \lambda_0$ where λ_0 are the parameters of the claimed speaker pdf, and the *alternative* hypothesis denoted by $H_1 : \lambda \neq \lambda_0$ which includes all possible values of λ which are not characteristic of the claimed speaker. Given a sample $O_1, \ldots O_n$ from a pdf $f(O;\lambda)$, we have to make a choice between H_0 and H_1. Rejecting H_0 when it is true is called a *type I error*, and accepting H_0 when it is false is a *type II error*. In classical hypothesis testing, the probability of making a type I error is set to some pre-specified value and a test is constructed which minimizes the probability of making a type II error. This corresponds to the *most powerful* test.

Unfortunately, designing a uniformly most powerful test in our situation is not possible because (1) the alternative hypothesis is composite ($\lambda \neq \lambda_0$); (2) the "true" pdf $f(O;\lambda)$ is unknown. To address this problem, the alternative composite hypothesis $H_1 : \lambda \neq \lambda_0$ can be replaced by a single hypothesis $H_1 : \lambda = \lambda_1$ where λ_1 is the parameters of the "anti-speaker" (sometimes called background or cohort model) of the claimed speaker. In practice, the parameters λ_0 and λ_1 of each pdf are estimated from 2 data sets, one containing data from the claimed (customer) speaker data, the other containing speech coming from other speakers. The estimation is usually performed using the maximum likelihood estimation criterion. The test is finally built using the Neyman-Pearson (NP) lemma or a generalized maximum likelihood ratio:

$$\text{Reject } H_0 \quad \text{if and only if} \quad \frac{f(O;\lambda_0)}{f(O;\lambda_1)} < \gamma. \tag{1}$$

Such a process is not optimal with respect to the verification task because: (1) the models λ_0 and λ_1 are separately trained using a criterion not related to the verification; (2) the NP lemma is used while its application is no longer valid (the "true" pdfs are unknown). Hence, it has been observed that the performance of speaker verification systems built through this process is closely related to the design of the anti-speaker model, usually carried out using "ad-hoc" rules [1].

In this paper, we propose a data-driven approach to automatically derive all parameters involved in the decision test (1), *i.e.* the speaker and anti-speaker models and the decision threshold. All parameters are optimized on a training data set according to a criterion directly related to the verification task. The idea is to approximate the total number of verification errors (type I *and* II) on the training corpus as a function of the different parameters using the minimum error training paradigm [2, 3]. Then, it becomes possible to minimize the number of errors with respect to these parameters.

Some recent reports on speaker identification/verification describe encouraging results obtained by using discriminative training formulations [4, 5]. In [4], minimum verification error training is used to minimize the number of type I errors on the training corpus. In [5], both types of error are minimized but the criterion used to build the models is not the same as the one used by the verification test, leading to a sub-optimal formulation.

In this paper, the customer and imposter (anti-speaker) models are built so that the number of verification errors (type I and II) on the training corpus is minimized. The criterion used to build the model is the same as the one used by the statistical test, leading to an optimal formulation. The proposed algorithm has been applied to build a fixed-password speaker verification system.

2. PRINCIPLE

Let $S^l = v_1^l, \ldots v_{N(l)}^l$ be the l^{th} string of $N(l)$ words used to verify the speaker identity. The different words are taken from a vocabulary set $\{w_k\}$, with $1 \leq k \leq K$. The speech segment associated with the word v_n^l is denoted O_n^l and is obtained from a speaker independent speech recognizer. The whole sentence is

denoted by O^l.

The verification rule (1) can be rewritten as a classification rule aiming at classifying the sentence O^l into two classes C_0 and C_1, associated with the hypotheses H_0, H_1 respectively:

$$\text{if} \quad L(O^l;\lambda_0) - L(O^l;\lambda_1) - \tau > 0 \quad \text{then} \quad O^l \in C_0 \quad (2)$$
$$\text{else} \quad O^l \in C_1,$$

where $L(\cdot)$ denotes the log-likelihood defined as $L(\cdot) = \log f(\cdot)$, and τ is the decision threshold.

Then, we can define a *misverification measure* for each class to count the verification errors based on the string score. The misverification functions are given by

$$d_0(O^l;\Lambda) = -L(O^l;\lambda_0) + L(O^l;\lambda_1) + \tau, \quad (3)$$
$$d_1(O^l;\Lambda) = +L(O^l;\lambda_0) - L(O^l;\lambda_1) - \tau = -d_0(O^l;\Lambda), \quad (4)$$

where $\Lambda = \{\lambda_0, \lambda_1, \tau\}$ is the set of parameters for the customer and imposter models and the decision threshold. The log-likelihoods $L(O^l;\lambda_0)$ and $L(O^l;\lambda_1)$ are calculated over the whole sentence S^l and normalized with respect to the total number of frames $M(l)$ in the sentence. Thus,

$$L(O^l;\lambda_0) = \frac{1}{M(l)} \sum_{n=1}^{N(l)} L(O_n^l; \lambda_0^{v_n^l}), \quad (5)$$

and similarly for $L(O^l;\lambda_1)$. The log-likelihood of the individual word segments O_n^l are derived using a Viterbi alignment against the customer and imposter models.

The two misclassification measures can be embedded into a smooth *loss function* so that the number of verification errors can be approximated. Two loss functions (one for each class) can be written:

$$l_0(O^l;\Lambda) = \frac{A_0}{1+\exp(-a d_0(O^l;\Lambda))} \quad \text{type I error}, \quad (6)$$

$$l_1(O^l;\Lambda) = \frac{A_1}{1+\exp(-a d_1(O^l;\Lambda))} \quad \text{type II error}. \quad (7)$$

The two constants A_0 and A_1 are used to specify different losses according to the type of error. Depending on the task, it is possible to emphasize type I or type II errors by modifying A_0 or A_1. The positive constant a is used to control the slope of the decision threshold.

The count of classification errors, for a given observation O^l is given by:

$$l(O^l;\Lambda) = l_0(O^l;\Lambda)1(O^l \in C_0) + l_1(O^l;\Lambda)1(O^l \in C_1) \quad (8)$$

where $1(\cdot)$ is the indicator function defined as:

$$1(O^l \in C_k) = \begin{cases} 1 & \text{if} \quad O^l \in C_k, \\ 0 & \text{otherwise}. \end{cases} \quad (9)$$

Finally, the *empirical average cost*, which is an approximation of the total number of verification errors (type I and II) on the training corpus $O^1, \ldots O^N$, can be obtained. The training corpus is partitioned into two sets, the first set containing the customer data $\mathcal{O}_0$, the second set the imposter data $\mathcal{O}_1$, we have:

$$L(\Lambda) = \sum_{l=1}^{N} l(O^l;\Lambda) \quad (10)$$
$$= \sum_{O^l \in \mathcal{O}_0} l_0(O^l;\Lambda) + \sum_{O^l \in \mathcal{O}_1} l_1(O^l;\Lambda). \quad (11)$$

Examining (11), it can be seen that the total number of verification errors (combination of type I and II) of the training corpus can be expressed as a continuous and differentiable function of the model parameters and the decision threshold. Consequently, it becomes possible to minimize (11) w.r.t. all parameters by using a gradient descent algorithm [3], and therefore get a joint estimate of all parameters involved in the verification task.

3. EXPERIMENTS AND RESULTS

3.1. System description

The speaker verification system is based on representing speakers using continuous density Gaussian mixture hidden Markov models (HMMs). The system is operated in a text-dependent mode, each customer having her/his own fixed digit-string password. Customers are represented by a set of HMM digit models comprising the digits in their password utterances. The customer digit models consist of 6 state left-to-right HMMs, with 4 Gaussian mixture components per state. These are trained using the segmental $\mathcal{K}$-means algorithm [6]. The background models contain the same number of states, but have 12 Gaussian mixtures per state. Digit-based segmentations are obtained from a separate speaker independent connected digit recognition system and are used to initialize model training and to specify the scoring segments of verification utterances.

For each digit segment, a Viterbi decoding is performed to find the optimal state segmentation of the associated digit. An average log-likelihood score is obtained over the whole sentence, excluding the silence segments. Average log-likelihood scores are obtained from the customer and background model, and a normalized verification score is computed by subtracting the background score from the customer score.

Two sets of experiments are carried out. In the first set, the goal is to compare the performance of the baseline system trained using Maximum Likelihood Estimation (MLE) to a system trained using Minimum Verification Error (MVE). Because most speaker verification systems are trained using a limited amount of data, adapting the system when new speech data are available leads to a significant improvement of performance. Hence, in the second set of experiments, the performance of two batch, supervised, adaptation scenarios is studied, one based on simplified maximum *a posteriori* (MAP) estimates, the other based on MVE. The initial set of models is the one built using MLE in the first set of experiments.

3.2. Database description

Experiments are performed using a fixed password speaker verification scenario. The database consists of 14-digit verification password utterances collected over the long distance telephone network. The evaluation set consists of 49 speakers (25 females and 24 males). Each of these speakers provided 3 enrollment utterances of the password, used as customer training data. Each speaker also

provided a series of verification utterances (customer test data), under various telephone and channel conditions. The average number of verification utterances is 65. In the adaptation experiments, 4 sentences used as customer adaptation data are omitted from the set of verification utterances. For each customer, a set of 30 or 35 speakers of the same gender each recorded 2 utterances of the customer's password. 10 imposter sentences for each customer are used as training data along with the customer training utterances for the MVE training/adaptation. The remaining imposter utterances are used as test data. The speech signal is parameterized using 12 linear predictive coding (LPC) derived cepstral coefficients [7] and their first derivatives. Cepstral mean normalization is applied to minimize channel variability.

3.3. Experimental Results

A priori thresholds are not assigned in our initial experiments. System performance is evaluated from pooled and averaged individual equal-error rates. Equal-error rate is calculated by sorting customer and imposter verification scores and finding the score value such that the fraction of customer scores less than that value is equal to the fraction of imposter scores greater than that value. This fraction is the equal-error rate, meaning that if the decision threshold is set to that score value, the false rejection rate is equal to the false acceptance rate. Two equal-error rates are calculated. The first one is an individual equal-error rate, which is averaged over all speakers. The second one is the pooled equal-error rate, obtained by pooling all customer scores and all imposter scores to obtain an equal-error rate based on a single threshold.

3.3.1. Comparison of MLE and MVE training

In MVE training, all model parameters (means, variances, mixture weights) are updated incrementally for each training utterance. No significant improvement is observed when adapting the decision threshold τ during the training so τ is set to 0 in all experiments. One utterance taken from the 3 customer training utterance is followed by one utterance taken from the 10 imposter utterances. One MVE iteration consists in updating the model using 6 utterances (3 customer + 3 imposter).

Table 1 shows the average individual equal-error rates calculated from customer models only (raw equal error rate), and the average individual equal-error rates calculated from the log-likelihood ratio of customer and imposter models (normalized equal error rate). It appears that the MVE training can reduce the raw equal-error rate by more than 25%, and the normalized equal-error rate by more than 20%.

It should be noted that a set of fixed variances is used to replace estimated model variances for both MLE customer and imposter models. However, the MLE models used as initial models for the MVE estimation are the ones with the original model variances. Using the fixed variances as initial variances for the MVE training leads to the same results.

3.3.2. Comparison of MAP and MVE adaptation

In the model adaptation experiments, the models trained using MLE are used as initial models. Two different model adaptations are performed in batch mode. The first one is an empirical MAP adaptation similar to the one described in Rosenberg and Soong [8] which uses 4 customer utterances to adapt the customer model, the background model being unchanged. The second one is a string-based

No adaptation		
Training criterion	Raw EER	Norm. EER
MLE	3.71	2.86
MVE	2.67	2.23

Table 1: Average individual equal-error rate (%) using raw (customer log-likelihood) scores and normalized (log-likelihood ratio) for MLE and MVE training.

After adaptation		
Adapt. criterion	Raw EER	Norm. EER
MAP	1.66	0.58
MVE	0.77	0.52

Table 2: Average raw and normalized individual equal-error rate (%) for MAP and MVE adaptation. Model variances are used in MAP and MVE adaptation.

MVE adaptation where the same 4 customer sentences plus 10 imposter sentences are used to adapt both the customer and imposter models.

Individual equal-error rates (EER's) are given Table 2 for both MAP and MVE adaptation. No statistically significant differences of the normalized EER can be observed between MVE and MAP adaptation. However, it can be seen that the EER calculated from raw scores is about 60% lower when applying MVE instead of MAP to adapt the models.

Another way of looking at the relation between MAP and MVE verification scores can be obtained by plotting the distribution of the scores for the customer and imposter sentences. The distributions of the pooled normalized scores are plotted in Fig 1. It appears that MVE adaptation tends to broaden the distributions and to separate the imposter and the customer scores, which is consistent with the MVE criterion.

Evaluating a verification system using equal-error rate does not address crucial practical issues. These issues are the setting of an *a priori* threshold τ and the sensitivity of the system to variations of the threshold. To appreciate this issue, the total error rate of a given speaker, defined as the sum of the false rejection and the false acceptance rate, *versus* the location of the decision threshold τ is plotted in Fig 2. (Similar plots can be obtained with other speakers.) This plot indicates that without any prior information about the dynamic range of the threshold location, the MVE models are less sensitive to the location of the threshold. This property of MVE adaptation over MAP can also be seen by plotting false rejection *versus* false acceptance rate for the pooled normalized scores in Fig 3. In this figure, we first observe that the MVE models exhibit a lower pooled equal-error rate (EER=1.68%) than the MAP model (EER=2.19%), (while the individual equal-error rates are similar). This curve also tends to indicate that a small variation of false rejection near the operating point leads to a variation of the false acceptance which is larger with MAP models than with MVE models.

In order to support this finding, an experiment was performed where the threshold is dynamically assigned. The idea is to calculate a new rejection threshold τ based on the current value of the threshold and the current verification score [9]. The experimental

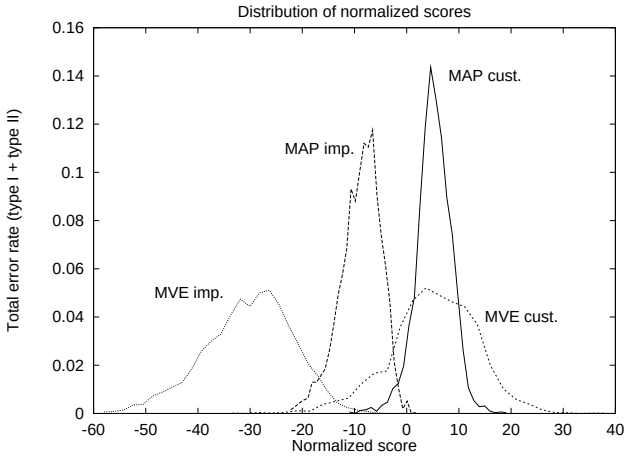

Figure 1: Histogram of normalized verification scores, pooled speakers, after MAP and MVE adaptation.

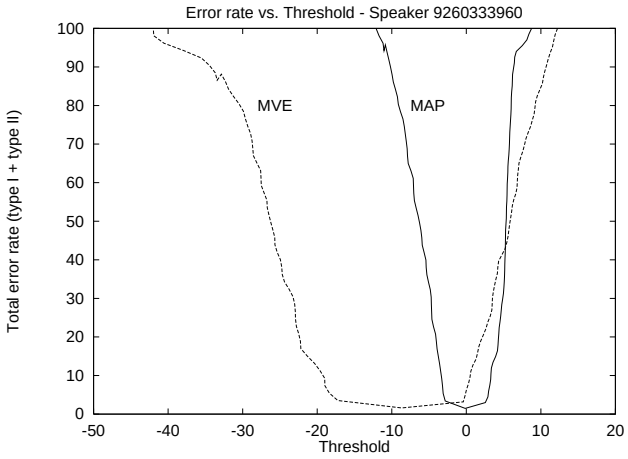

Figure 2: Total error rate (false rejection + false acceptance) *vs.* threshold location. Speaker 9260333960.

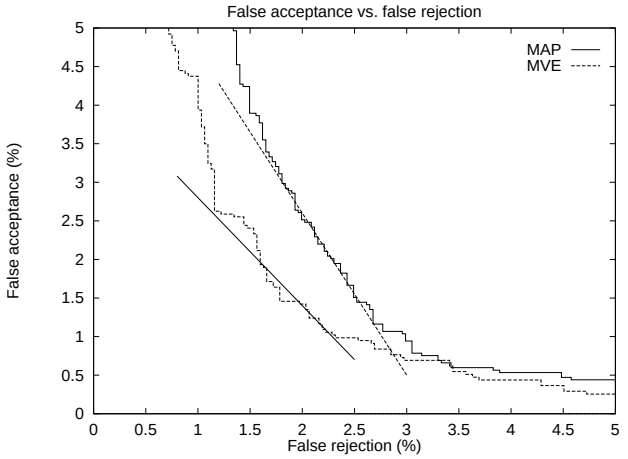

Figure 3: False rejection *vs.* false acceptance, pooled speakers, after MAP and MVE adaptation.

dynamic threshold assignment				
Adapt. criterion	F. rej.	F. acc.	Raw EER	Norm. EER
MAP	3.52	2.70	1.57	0.64
MVE	1.90	1.39	0.79	0.57

Table 3: Average individual False rejection (%), false acceptance (%), raw equal-error rate (%) and normalized equal-error rate (%) after dynamic threshold assignment for MVE and MAP adaptation.

conditions for calculating the false rejection and false acceptance rates are that the customer utterances are processed first to adapt the threshold (from utterance to utterance) and then all the imposter utterances are evaluated with the threshold obtained from the last customer utterance. This speaker dependent threshold assignment scenario is independent of and unrelated to the model training criterion. The false rejection and false acceptance rates obtained with speaker dependent threshold assignments are given for both MAP and MVE adaptation in Table 3 along with the raw and normalized individual equal-error rates. When using MVE adaptation, the false rejection rate drops from 3.52% to 1.90%, and the false rejection rate drop from 2.70% to 1.39%, a decrease close to 45% in both false rejection and false acceptance. This result suggests that while MVE and MAP adaptations lead to similar individual equal-error rates, the verification system built from the MVE criterion exhibits less sensitivity to the threshold setting, and is therefore a more robust system than the one built using MAP.

4. CONCLUSION

We have proposed an algorithm to automatically build a speaker verification system by minimizing the total number of string verification errors on the training corpus. Speaker and anti-speaker models are simultaneously estimated according to a criterion related to the verification task, leading to an optimal data-driven formulation. Experiments on a fixed password speaker verification task have suggested that the MVE training can effectively improve the separation between speaker and anti-speaker models and can lead to a system less sensitive to the setting of the verification threshold.

5. REFERENCES

[1] A. E. Rosenberg and S. Parthasarathy. Speaker background models for connected digit password speaker verification. In *Proc. IEEE ICASSP*, volume 1, pages 81–84, 1996.

[2] A. Ljolje, Y. Ephraim, and L. R. Rabiner. Estimation of hidden Markov model parameters by minimizing empirical error rate. In *Proc. IEEE ICASSP*, pages 709–712, Albuquerque, New Mexico, April 1990.

[3] B.-H. Juang and S. Katagiri. Discriminative learning for minimum error classification. *IEEE Transactions on Signal Processing*, 40(12):3043–3054, 1992.

[4] C.-S. Liu, C.-H. Lee, W. Chou, B.-H. Juang, and A. E. Rosenberg. A study on minimum error discriminative training for speaker recognition. *Journal of the Acoustical Society of America*, 97(1):637–648, January 1995.

[5] F. Korkmazskiy and B.-H. Juang. Discriminative adaptation for speaker verification. In *Proc. ICSLP*, volume 3, pages 28–31, Philadelphia, USA, 1996.

[6] L. R. Rabiner, J. G. Wilpon, and B.-H. Juang. A segmental K-means training procedure for connected word recognition. *AT&T Bell Labs Tech. J.*, 65(3):21–31, 1986.

[7] L.R. Rabiner and B.-H. Juang. *Fundamentals of speech recognition*, chapter 3. Siganl processing. Prentice Hall, 1993.

[8] A. E. Rosenberg and F. K. Soong. Evaluation of a vector quantization talker recognition system in text independent and text dependent modes. *Computer Speech and Language*, 22:143–157, 1987.

[9] A. E. Rosenberg and S. Parthasarathy. Personal communication, 1996.