IMPROVED METHODS FOR CHARACTERIZING THE ALTERNATIVE HYPOTHESIS USING MINIMUM VERIFICATION ERROR TRAINING FOR LLR-BASED SPEAKER VERIFICATION

Yi-Hsiang Chao[1,2], Wei-Ho Tsai[3], Hsin-Min Wang[1] and Ruei-Chuan Chang[1,2]

[1] Institute of Information Science, Academia Sinica, Taipei, Taiwan
[2] Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan
[3] Department of Electronic Engineering, National Taipei University of Technology, Taipei, Taiwan
{yschao,whm}@iis.sinica.edu.tw, whtsai@en.ntut.edu.tw, rc@cc.nctu.edu.tw

ABSTRACT

Speaker verification based on the log-likelihood ratio (LLR) is essentially a task of modeling and testing two hypotheses: the null hypothesis and the alternative hypothesis. Since the alternative hypothesis involves unknown imposters, it is usually hard to characterize a priori. In this paper, we propose a framework to better characterize the alternative hypothesis with the goal of optimally separating client speakers from imposters. The proposed framework is built on either a weighted arithmetic combination or a weighted geometric combination of useful information extracted from a set of pre-trained anti-speaker models. The parameters associated with the combinations are then optimized using Minimum Verification Error training such that both the false acceptance probability and the false rejection probability are minimized. Our experiment results show that the proposed framework outperforms conventional LLR-based approaches.

Index Terms— Speaker recognition, minimization methods, hypothesis testing, minimum verification error.

1. INTRODUCTION

Speaker verification is usually formulated as a statistical hypothesis testing problem and solved using a log-likelihood ratio (LLR) test [1]. Given an input utterance U, the LLR test for determining whether or not U is spoken by the hypothesized speaker is performed as follows

$$L(U) = \log \frac{p(U \mid H_0)}{p(U \mid H_1)} \begin{cases} \geq \theta & \text{accept } H_0 \\ < \theta & \text{accept } H_1 \text{ (reject } H_0 \text{)}, \end{cases} \tag{1}$$

where H_0 represents that U is spoken by the hypothesized speaker (called the *null hypothesis*); H_1 represents that U is not spoken by the hypothesized speaker (called the *alternative hypothesis*); $p(U \mid H_i)$, i = 0 or 1, is the likelihood of hypothesis H_i given utterance U; and θ is a decision threshold. In practical implementations, H_0 and H_1 are usually characterized by some parametric models, such as Gaussian mixture models (GMMs) [1]. However, even though H_0 can be modeled straightforwardly using speech utterances from the hypothesized speaker, H_1 does not involve any specific speaker, and hence lacks explicit data for modeling. Thus, a number of approaches have been proposed to better characterize H_1. The common strategy is to generate one or multiple models using speech from a large number of non-hypothesized speakers, and then compute the likelihood $P(U \mid H_1)$ using [2]:

$$p(U \mid H_1) = \Psi\big(p(U \mid \lambda_1), p(U \mid \lambda_2) ..., p(U \mid \lambda_N)\big), \tag{2}$$

where $\Psi(\cdot)$ denotes a certain function of the likelihoods computed for a set of *background models* $\{\lambda_1, \lambda_2, ..., \lambda_N\}$ representing the potential imposters. For example, if $\Psi(\cdot)$ is an arithmetic mean [1], the LLR is of the form

$$L_1(U) = \log p(U \mid \lambda) - \log\left\{\frac{1}{N}\sum_{i=1}^{N} p(U \mid \lambda_i)\right\}, \tag{3}$$

where λ denotes a model generated for the hypothesized speaker. Alternatively, the arithmetic mean can be replaced by a maximum function [4], which yields the LLR

$$L_2(U) = \log p(U \mid \lambda) - \max_{1 \leq i \leq N} \log p(U \mid \lambda_i), \tag{4}$$

or by a geometric mean [5], which yields the LLR

$$L_3(U) = \log p(U \mid \lambda) - \frac{1}{N}\sum_{i=1}^{N} \log p(U \mid \lambda_i). \tag{5}$$

A special case arises when N = 1, where a single background model is usually trained by pooling all the available data; this is called a *world model* [2]. The LLR in this case becomes

$$L_4(U) = \log p(U \mid \lambda) - \log p(U \mid \Omega), \tag{6}$$

where Ω denotes the world model.

However, there is no theoretical evidence to indicate which method of characterizing H_1 is optimal, and the selection of $\Psi(\cdot)$ is usually application and training data dependent. In particular, a simple function, such as the arithmetic mean, the maximum, or the geometric mean, is a heuristic that does not involve an optimization process. Thus, the resulting system is far from optimal in terms of verification accuracy. To better handle this problem, we propose a framework that characterizes the alternative hypothesis by exploiting information available from background models, such that utterances from the imposters and the hypothesized speaker can be separated more effectively. The framework is built on either a weighted geometric combination or a weighted arithmetic combination of the likelihoods computed for background models. In contrast to the geometric mean $L_3(U)$ or the arithmetic mean $L_1(U)$, which are independent of the system training, our combination scheme treats the background models unequally according to how close each individual is to the hypothesized speaker model, and quantifies the unequal nature of the background models by a set of weights optimized in the training phase. The optimization is carried out by Minimum

Verification Error (MVE) training [6,7], which minimizes both the false acceptance probability and the false rejection probability.

The remainder of the paper is organized as follows. Section 2 introduces the proposed methods for characterizing the alternative hypothesis. Section 3 describes an MVE training method used to optimize our methods. Section 4, contains the experiment results. Finally, in Section 5, we present our conclusions.

2. CHARACTERIZATION OF THE ALTERNATIVE HYPOTHESIS

Instead of using the heuristic arithmetic mean or geometric mean, our goal is to design a function $\Psi(\cdot)$ that optimally exploits the information available from background models. This section presents our design approach, which is based on either the weighted arithmetic combination or the weighted geometric combination of the useful information available.

2.1. The Weighted Arithmetic Combination (WAC)

The weighted arithmetic combination is defined as

$$p(U \mid H_1) = \Psi(p(U \mid \lambda_1), ..., p(U \mid \lambda_N)) = \sum_{i=1}^{N} w_i \, p(U \mid \lambda_i), \quad (7)$$

where w_i is the weight of the likelihood $p(U \mid \lambda_i)$ subject to $\sum_{i=1}^{N} w_i = 1$. This function assigns different weights to N background models to indicate their individual contribution to the alternative hypothesis. Suppose all the N background models are Gaussian Mixture Models (GMMs). Eq. (7) constitutes a two–layer structure of a GMM, in which one layer represents each background model and the other represents the combination of background models.

2.2. The Weighted Geometric Combination (WGC)

Alternatively, we can define the function $\Psi(\cdot)$ in Eq. (2) from the perspective of the weighted geometric combination as

$$p(U \mid H_1) = \Psi(p(U \mid \lambda_1), ..., p(U \mid \lambda_N)) = \prod_{i=1}^{N} p(U \mid \lambda_i)^{w_i} \, . \quad (8)$$

Similar to the weighted arithmetic combination, Eq. (8) considers the individual contribution of a background model to the alternative hypothesis by assigning a weight to each likelihood probability. One additional advantage of WGC is that it avoids the problem where $p(U \mid H_1) = 0$, which could happen with the heuristic geometric mean because some values of the likelihood may be rather small when the background models are irrelevant to an input utterance U. With a weight attached to each background model, $\Psi(\cdot)$ defined in Eq. (8) should be less sensitive to a tiny value of the likelihood; hence, it should be more robust than the heuristic geometric mean.

2.3. Relations to the conventional LLRs

We observe that Eq. (7) and Eq. (8) are equivalent to the arithmetic mean in Eq. (3) and the geometric mean in Eq. (5), respectively, when $w_i = 1/N$, $i = 1,2,\ldots, N$, i.e., all the background models are assumed to contribute equally. It can also be observed that both Eq. (7) and Eq. (8) will degenerate to a maximum function in Eq. (4), if we set $w_i = 0$, $\forall$ i, except $w_{i^*} = 1$, where $i^* = \arg\max_{1 \le i \le N} p(U \mid \lambda_i)$. Furthermore, Eqs. (7) and (8) will degenerate to $L_4(U)$ in Eq. (6), if only a world model Ω is used as the background model. Thus, both WAC and WGC can be viewed as generalized and trainable versions of $L_1(U)$, $L_2(U)$, $L_3(U)$ or $L_4(U)$.

3. MINIMUM VERIFICATION ERROR TRAINING

After representing $\Psi(\cdot)$ as a trainable combination of likelihoods, the task is reduced to solving the associated weights. To obtain an optimal set of weights, we propose using Minimum Verification Error (MVE) training [6,7].

In order for the decision threshold θ to be included in the optimization, we express Eq. (1) as the following equivalent test

$$L(U) = \log p(U \mid \lambda) - \log p(U \mid H_1) - \theta \begin{cases} \ge 0 & \text{accept } H_0 \\ < 0 & \text{accept } H_1 \, , \end{cases} \quad (9)$$

and define a mis-verification measure

$$d(U) = \begin{cases} -L(U) & \text{if } U \in H_0 \\ L(U) & \text{if } U \in H_1 \, . \end{cases} \quad (10)$$

The measure is then converted into a value between 0 and 1 using a sigmoid function $s(d(U)) = 1/[1+\exp(-a \cdot d\,(U))]$, where a is a scalar, so that it reflects the verification error probability. Next, a loss function $\ell_i(U)$, $i = 0$ or 1, is used to describe the average false rejection errors ($i = 0$) or false acceptance errors ($i = 1$):

$$\ell_i(U) = \frac{1}{N_i} \sum_{U \in H_i} s(d(U)), \quad (11)$$

where N_0 and N_1 are the numbers of utterances from true speakers and impostors, respectively. Finally, an overall expected loss is defined by

$$D(U) = x_0 \ell_0(U) + x_1 \ell_1(U), \quad (12)$$

where x_0 and x_1 reflect which type of error is of more concern than the other in a practical application.

Accordingly, our goal is to find the weights w_i in Eq. (7) and Eq. (8) such that Eq. (12) is minimized. This can be achieved by using the Gradient Probabilistic Descent (GPD) method [6]. To ensure that the weights satisfy $\sum_{i=1}^{N} w_i = 1$, we solve w_i by means of an intermediate parameter α_i, where $w_i = \exp(\alpha_i) / \sum_j \exp(\alpha_j)$, similar to the strategy used in [6]. Parameter α_i is iteratively optimized using

$$\alpha_i^{(k+1)} = \alpha_i^k - \eta \frac{\partial D(U)}{\partial \alpha_i}, \quad (13)$$

where η is the step size, and

$$\begin{aligned} \frac{\partial D(U)}{\partial \alpha_i} &= x_0 \frac{\partial \ell_0(U)}{\partial \alpha_i} + x_1 \frac{\partial \ell_1(U)}{\partial \alpha_i} \\ &= x_0 \frac{\partial \ell_0}{\partial s} \cdot \frac{\partial s}{\partial d} \cdot \frac{\partial d}{\partial L} \cdot \frac{\partial L}{\partial w_i} \cdot \frac{\partial w_i}{\partial \alpha_i} + x_1 \frac{\partial \ell_1}{\partial s} \cdot \frac{\partial s}{\partial d} \cdot \frac{\partial d}{\partial L} \cdot \frac{\partial L}{\partial w_i} \cdot \frac{\partial w_i}{\partial \alpha_i} \\ &= x_0 \cdot \frac{1}{N_0} \sum_{U \in H_0} \left\{ a \cdot s(-L(U))[1 - s(-L(U))] \cdot - \frac{\partial L}{\partial w_i} \cdot w_i (1 - w_i) \right\} \\ &\quad + x_1 \cdot \frac{1}{N_1} \sum_{U \in H_1} \left\{ a \cdot s(L(U))[1 - s(-L(U))] \cdot \frac{\partial L}{\partial w_i} \cdot w_i (1 - w_i) \right\} . \end{aligned} \quad (14)$$

If WAC is used, then

$$\frac{\partial L}{\partial w_i} = \frac{-\partial}{\partial w_i}\log\left(\sum_j w_j\, p(U\,|\,\lambda_j)\right) = \frac{-p(U\,|\,\lambda_i)}{\sum_j w_j\, p(U\,|\,\lambda_j)}. \tag{15}$$

If WGC is used, then

$$\frac{\partial L}{\partial w_i} = \frac{-\partial}{\partial w_i}\left(\sum_j w_j \log p(U\,|\,\lambda_j)\right) = -\log p(U\,|\,\lambda_i). \tag{16}$$

The threshold θ in Eq. (9) can be estimated using [7]:

$$\theta^{(k+1)} = \theta^k - \eta\frac{\partial D(U)}{\partial \theta}, \tag{17}$$

where

$$\begin{aligned}\frac{\partial D(U)}{\partial \theta} &= x_0 \frac{\partial \ell_0}{\partial s}\cdot\frac{\partial s}{\partial d}\cdot\frac{\partial d}{\partial L}\cdot\frac{\partial L}{\partial \theta} + x_1 \frac{\partial \ell_1}{\partial s}\cdot\frac{\partial s}{\partial d}\cdot\frac{\partial d}{\partial L}\cdot\frac{\partial L}{\partial \theta} \\ &= x_0\cdot\frac{1}{N_0}\sum_{U\in H_0} a\cdot s(-L(U))\big[1-s(-L(U))\big] \\ &\quad - x_1\cdot\frac{1}{N_1}\sum_{U\in H_1} a\cdot s(L(U))\big[1-s(-L(U))\big].\end{aligned} \tag{18}$$

In our implementation, the overall expected loss is set according to the Detection Cost Function (DCF) [8]:

$$\begin{aligned}C_{DET} &= C_{Miss}\times P_{Miss}\times P_{T\arg et} \\ &\quad + C_{FalseAlarm}\times P_{FalseAlarm}\times(1-P_{T\arg et}),\end{aligned} \tag{19}$$

where P_{Miss} is the miss (false rejection) probability, $P_{FalseAlarm}$ is the false alarm (false acceptance) probability, P_{Target} is the *a priori* probability of the target (hypothesized) speaker, and C_{Miss} and $C_{FalseAlarm}$ are the relative costs of the missed error and false alarm error, respectively. A special case of DCF is known as the Half Total Error Rate (HTER), where C_{Miss} and $C_{FalseAlarm}$ are both equal to 1, and P_{Target} = 0.5, i.e., HTER = (P_{Miss} + $P_{FalseAlarm}$) / 2. Then, approximating P_{Miss} and $P_{FalseAlarm}$ by $\ell_0(U)$ and $\ell_1(U)$, respectively, we set the overall expected loss specifically as

$$\begin{aligned}D(U) &= C_{Miss}\times \ell_0(U)\times P_{T\arg et} \\ &\quad + C_{FalseAlarm}\times \ell_1(U)\times(1-P_{T\arg et}).\end{aligned} \tag{20}$$

4. EXPERIMENTS

4.1. Experiment setup

We conducted speaker-verification experiments on speech data extracted from the XM2VTSDB multi-modal database [10]. In accordance with "Configuration II" described in [10], the database was divided into three subsets: "Training", "Evaluation", and "Test". We used "Training" to build each client model and the background models, and used "Evaluation" to optimize the weights w_i in Eq. (7) or Eq. (8), along with the threshold θ. Then, the speaker verification performance was evaluated on "Test". As shown in Table 1, a total of 293 speakers[1] in the database were divided into 199 clients, 25 "evaluation impostors", and 69 "test impostors". Each speaker participated in 4 recording sessions at about one-month intervals, and each recording session consisted of 2 shots. In each shot, the speaker was prompted to utter 3 sentences "0 1 2 3 4 5 6 7 8 9", "5 0 6 9 2 8 1 3 7 4", and "Joe took father's green shoe bench out". Each utterance, sampled at 32 kHz, was converted into a stream of 24-order feature vectors, each consisting of 12 Mel-scale frequency cepstral coefficients [11] and their first time derivatives, by a 32-ms Hamming-windowed frame with 10-ms shifts.

Table 1. Configuration of the speech database.

Session	Shot	199 clients	25 impostors	69 impostors
1	1	Training	Evaluation	Test
	2			
2	1			
	2			
3	1	Evaluation		
	2			
4	1	Test		
	2			

We used 12 (2×2×3) utterances/speaker from sessions 1 and 2 to train each client model, represented by a GMM with 64 mixture components. For each client, the other 198 clients' utterances from sessions 1 and 2 were used to generate the world model, represented by a GMM with 256 mixture components. Meanwhile, B speakers were chosen from these 198 clients as the cohort [3] to yield B background models. Then, to optimize the weights, w_i, and the threshold, θ, we used 6 utterances/client from session 3, along with 24 (4×2×3) utterances/evaluation-impostor over the four sessions, which yielded 1,194 (6×199) client samples and 119,400 (24×25×199) impostor samples. In the performance evaluation, we tested 6 utterances/client in session 4 and 24 utterances/test-impostor over the four sessions, which involved 1,194 (6×199) client trials and 329,544 (24×69×199) impostor trials.

In addition, we used the B cohort set of models for $L_1(U)$ in Eq. (3), $L_2(U)$ in Eq. (4), and $L_3(U)$ in Eq. (5), and B+1 background models, consisting of the B cohort set of models and one world model for our WAC and WGC methods. B was empirically set to 20. Two cohort selection methods [1] were used. One selected the closest B speakers for each client; and the other selected the closest B/2 speakers, plus the farthest B/2 speakers for each client. Here, the degree of closeness is measured in terms of the pairwise distance defined by [1]:

$$d(\lambda_i,\lambda_j) = \log\frac{p(X_i\,|\,\lambda_i)}{p(X_i\,|\,\lambda_j)} + \log\frac{p(X_j\,|\,\lambda_j)}{p(X_j\,|\,\lambda_i)}, \tag{21}$$

where λ_i and λ_j are speaker models trained using the i-th speaker's utterances X_i and the j-th speaker's utterances X_j, respectively.

4.2. Experiment results

The proposed weighted combination methods were implemented in three ways: 1) WAC with the world model and the 10 closest cohort models, plus the 10 farthest cohort models ("WAC_w_10c_10f"); 2) WAC with the world model plus the 20 closest cohort models ("WAC_w_20c"); and 3) WGC with the world model plus the 20 closest cohort models ("WGC_w_20c"). The MVE training for both WAC and WGC was initialized with an equal weight, w_i, and the threshold θ was set to 0. The overall expected loss function D in Eq. (20) was set according to the HTER with C_{Miss} = 1, $C_{FalseAlarm}$ = 1, and P_{Target} = 0.5.

For the performance comparison, we used five systems as our baselines: 1) $L_1(U)$ with the 10 closest cohort models plus the 10

[1] We discarded 2 speakers (ID numbers 313 and 342) because of partial data corruption.

farthest cohort models ("$L1$_10c_10f"); 2) $L_1(U)$ with the 20 closest cohort models ("$L1$_20c"); 3) $L_2(U)$ with the 20 closest cohort models ("$L2$_20c"); 4) $L_3(U)$ with the 20 closest cohort models ("$L3$_20c"); and 5) $L_4(U)$ ("$L4$").

Fig. 1 shows the DET curves [9] for the speaker verification performance achieved by various methods. For each baseline, the value of the decision threshold θ was tuned to minimize HTER on "Evaluation", and then applied to "Test". The decision thresholds of the proposed methods were optimized automatically using "Evaluation", and then applied to "Test". From Fig. 1, we observe that both the proposed methods, WAC and WGC, outperform all the baseline systems. It can also be seen that the performance of WAC is slightly better than that of WGC, while there is no significant difference between "WAC_w_10c_10f" and "WAC_w_20c". Table 2 summarizes the experiment results based on HTER. Finally, the table shows that each of the proposed methods achieved a relative improvement of more than 10% over the best baseline system, "$L1$_10c_10f".

6. CONCLUSION

We have proposed a framework to improve the characterization of the alternative hypothesis for speaker verification. The framework is built on either a weighted arithmetic combination (WAC) or a weighted geometric combination (WGC) of useful information extracted from a set of pre-trained anti-speaker models. The parameters associated with the combinations are then optimized using Minimum Verification Error training such that both the false acceptance probability and the false rejection probability are minimized. Our experiment results demonstrate that the proposed framework outperforms conventional LLR-based approaches. In the future, we will study different optimization methods, such as boosting algorithms [12], to solve the weights in WAC and WGC. We will also evaluate the proposed framework on different applications related to user verification.

7. ACKNOWLEDGEMENTS

This work was funded in part by the National Science Council, Taiwan, under Grant: NSC94-2213-E-001-009.

8. REFERENCES

[1] D. A. Reynolds, "Speaker Identification and Verification Using Gaussian Mixture Speaker Models", *Speech Communication*, vol.17, pp. 91-108, 1995.

[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] A. E. Rosenberg, J. Delong, C. H. Lee, B. H. Juang and F. K. Soong, "The Use of Cohort Normalized Scores for Speaker Verification", *Proc. ICSLP1992*.

[4] A. Higgins, L. Bahler, and J. Porter, "Speaker Verification Using Randomized Phrase Prompting", *Digital Signal Processing*, vol. 1, no. 2, pp. 89-106, 1991.

[5] C. S. Liu, H. C. Wang, and C. H. Lee, "Speaker Verification Using Normalized Log-Likelihood Score", *IEEE Trans. Speech and Audio Processing*, vol. 4, pp. 56-60, 1996.

[6] W. Chou and B. H. Juang, *Pattern Recognition in Speech and Language Processing*, CRC Press, 2003.

[7] A. E. Rosenberg, O. Siohan, and S. Parthasarathy, "Speaker Verification Using Minimum Verification Error Training", *Proc. ICASSP1998*.

[8] http://www.nist.gov/speech/tests/spk/index.htm

[9] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET Curve in Assessment of Detection Task Performance", *Proc. Eurospeech1997*.

[10] J. Luettin and G. Maître, *Evaluation Protocol for the Extended M2VTS Database (XM2VTSDB)*, IDIAP-COM 98-05, IDIAP, 1998.

[11] X. Huang, A. Acero, and H. W. Hon, *Spoken Language Processing*, Prentics Hall, New Jersey, 2001.

[12] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd. ed., Springer, New York, 2001.

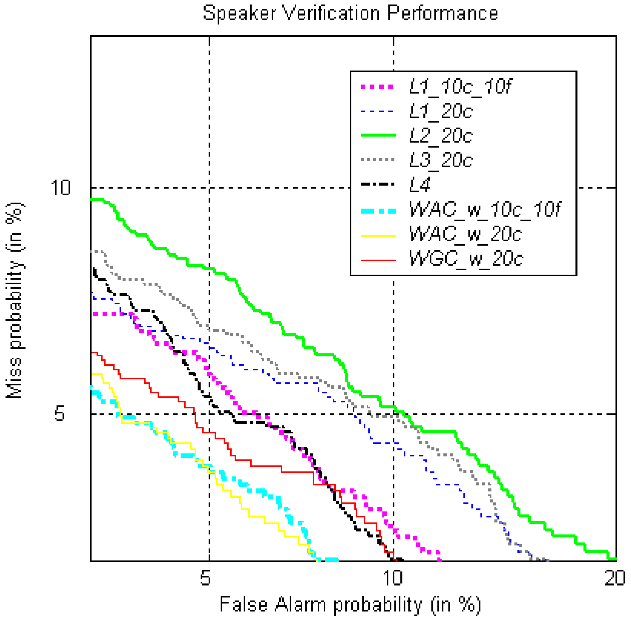

Fig. 1. DET curves.

Table 2. Experiment results in terms of HTER.

Methods	HTER
$L1$_10c_10f	0.0515
$L1$_20c	0.0535
$L2$_20c	0.0635
$L3$_20c	0.0583
$L4$	0.0519
WAC_w_10c_10f	0.0457
WAC_w_20c	0.0443
WGC_w_20c	0.0470