Kernel-Based Probabilistic Neural Networks with Integrated Scoring Normalization for Speaker Verification

Kwok-Kwong Yiu[1], Man-Wai Mak[1], and Sun-Yuan Kung[2⋆]

[1] Center for Multimedia Signal Processing
Dept. of Electronic and Information Engineering
The Hong Kong Polytechnic University, China
[2] Dept. of Electrical Engineering
Princeton University
USA

Abstract. This paper investigates kernel-based probabilistic neural networks for speaker verification in clean and noisy environments. In particular, it compares the performance and characteristics of speaker verification systems that use probabilistic decision-based neural networks (PDBNNs), Gaussian mixture models (GMMs) and elliptical basis function networks (EBFNs) as speaker models. Experimental evaluations based on 138 speakers of the YOHO corpus and its noisy variants were conducted. The original PDBNN training algorithm was also modified to make PDBNNs appropriate for speaker verification. Experimental evaluations, based on 138 speakers and the visualization of decision boundaries, indicate that GMM- and PDBNN-based speaker models are superior to the EBFN ones in terms of performance and generalization capability. This work also finds that PDBNNs and GMMs are more robust than EBFNs in verifying speakers in noise environments.

1 Introduction

Speaker verification aims to verify the validity of a claimed identity through voice. Text-dependent approaches, such as dynamic time warping (DTW) and hidden Markov models (HMMs) [1], explore the static and temporal characteristics of speakers. On the other hand, text-independent approaches, such as vector quantization (VQ) [2] and Gaussian mixture models (GMM) [3], assume independence among feature vectors and make use of distortion measures or probabilistic estimates. Most of these approaches, however, use data from the target speakers only to train the speaker models. As a result, discriminative information from anti-speakers will not be embedded in the speaker models.

⋆ This work was supported by The Hong Kong Polytechnic University, Grant No. G-W076. S. Y. Kung is on sabbatical from Princeton University. He is currently with The Hong Kong Polytechnic University.

Discriminative information can be utilized during model training and evaluation. For the former, supervised learning algorithms are used to discriminate within-class data from out-of-class data. For the latter, likelihood ratio [4] or scoring normalization [5] are applied during evaluation.

Neural networks are one of the approaches that allow discriminative information to be embedded in the speaker models. For example, the elliptical basis function networks proposed in [6] include the cluster centers of anti-speakers' speech in their hidden layer. It was shown that EBFNs perform better than radial basis function networks (RBFNs) and VQ. The neural tree networks (NTNs) are another type of networks that use discriminative training, and research has shown that NTNs are superior to VQ in speaker recognition tasks [7].

One of the main challenges in speaker recognition is to recognize speakers in adverse conditions. Noise is commonly considered as additive components to the speech signals. Speaker models trained by using clean speech signals are usually subject to performance degradation in noisy environments. The present study compares the speaker verification performance of three kernel-based speaker models under clean and noisy environments. They are Gaussian Mixture Models (GMMs), Elliptical Basis Function Networks (EBFNs) and Probabilistic Decision-Based Neural Networks (PDBNNs) [8]. The comparison aims to demonstrate the effect of supervised learning on the speaker models (least squares learning on EBFNs and reinforced learning on PDBNNs). For example, by comparing GMMs against PDBNNs, the importance of reinforced learning can be highlighted.

Three problem sets have been used in this study. These include a large-scale speaker verification experiment, speaker classification based on 2-D speech features and speaker verification using noisy variants of the YOHO corpus.

2 Speech Corpus and Pre-Processing

The YOHO corpus [9] was collected by ITT Defense Communication Division. The corpus features "combination lock" phrases, 138 speakers (108 male, 30 female), inter-session variability, and high-quality telephone speech (3.8kHz/clean). These features make YOHO ideal for speaker verification research. In this work, Gaussian white noise with different noise power was added to the clean YOHO corpus. Both the clean and noisy YOHO corpora were used in the experimental evaluations.

LP-derived cepstral coefficients were used as acoustic features. For each utterance, the silent regions were removed, and the remaining signals were pre-emphasized. Twelfth-order LP-derived cepstral coefficients were then computed using a 28 ms Hamming window at a frame rate of 14 ms.

3 Enrollment Procedures

Each registered speaker was assigned a personalized network (GMM, EBFN or PDBNN) modeling the characteristics of his/her own voice. Each network was trained to recognize the speech derived from two classes—speaker class and anti-speaker class. To this end, two groups of kernel functions (one group representing

the speaker himself/herself while the other representing the speakers in the anti-speaker class) were assigned to each network. We denote the group corresponding to the speaker class as the speaker kernels and the one corresponding to the anti-speaker class as the anti-speaker kernels. For each registered speaker, a unique anti-speaker set containing 16 anti-speakers was created. This set was used to create the anti-speaker kernels. The anti-speaker kernels enable us to integrate scoring normalization [10] into the networks, which enhances the networks' capability in discriminating the true speakers from the impostors.

4 Verification Procedures

Verification was performed using each speaker in the YOHO corpus as a claimant, with 64 impostors being randomly selected from the remaining speakers (excluding the anti-speakers and the claimant) and rotating through all the speakers. For each claimant, the feature vectors of the claimant's utterances from his/her 10 verification sessions in YOHO were concatenated to form a claimant sequence. Likewise, the feature vectors of the impostor's utterances were concatenated to form an impostor sequence.

The feature vectors from the claimant's speech $\mathcal{T}^c = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{T_c}\}$ was divided into a number of overlapping segments containing $T(< T_c)$ consecutive vectors. For the t-th segment ($\mathcal{T}_t \subset \mathcal{T}^c$), the average normalized log-likelihood

$$z_t = \frac{1}{T} \sum_{\boldsymbol{x} \in \mathcal{T}_t} \{\phi_S(\boldsymbol{x}) - \phi_A(\boldsymbol{x})\} \tag{1}$$

of the PDBNN and GMM speaker models was computed, where $\phi_S(\boldsymbol{x})$ and $\phi_A(\boldsymbol{x})$ represents the log-likelihood function of the speaker and anti-speaker respectively [8]. Verification decisions were based on the criterion:

$$\text{If } z_t \begin{cases} > \zeta & \text{accept the claimant} \\ \leq \zeta & \text{reject the claimant} \end{cases} \tag{2}$$

where ζ is a speaker-dependent decision threshold (see Section 5 below for the procedure of determining ζ). A verification decision was made for each segment, with the error rate (either FAR or FRR) being the proportion of incorrect verification decisions to the total number of decisions. In this work, T in Eqn. (1) was set to 500 (i.e., 7 seconds of speech), and each segment was separated by five consecutive vectors.

For the EBFN-based speaker models, verification decisions were based on the difference between the scaled network outputs [6]. Again, computing the difference between the two outputs is equivalent to normalizing the score in GMMs. Thus, we integrate scoring normalization into the network architecture.

5 Threshold Determination

The procedures for determining the decision thresholds of PDBNNs, GMMs and EBFNs are different. For GMM and EBFN speaker models, the utterances

from all enrollment sessions of 16 randomly selected anti-speakers were used for threshold determination [11]. Specifically, these utterances were concatenated and the procedure described in Section 4 was applied. The threshold ζ was adjusted until the FAR fell below a pre-defined level. In this work, we set this level to 0.5%.

To adopt PDBNNs to speaker verification, three modifications on the PDBNN's training algorithm have been made. First, we modified the likelihood computation such that only one threshold per speaker is required. Specifically, instead of comparing the network's loglikelihood against its corresponding threshold as in the original PDBNNs, we compared a normalized score against a single decision threshold as in Eqns. (1) and (2).

In the second modification, we changed the frequency at which the threshold is updated. As our speaker verification procedure is based on a segmental mode (see Section 4), we modified the globally supervised training to work on a segmental mode as follows. Let $\mathcal{T}_n$ be the n-th segment extracted from speaker's speech patterns $\mathcal{X}_S$ or from anti-speakers' speech patterns $\mathcal{X}_A$, the normalized segmental score is computed by evaluating

$$S(\mathcal{T}_n) = S_S(\mathcal{T}_n) - S_A(\mathcal{T}_n) = \frac{1}{T} \sum_{\boldsymbol{x} \in \mathcal{T}_n} \{\phi_S(\boldsymbol{x}) - \phi_A(\boldsymbol{x})\}.$$

For each segment, a verification decision was made according to the criterion:

$$\text{If } S(\mathcal{T}_n) \begin{cases} > \zeta_{n-1}^{(j)} & \text{accept the claimant} \\ \leq \zeta_{n-1}^{(j)} & \text{reject the claimant} \end{cases} \tag{3}$$

where $\zeta_{n-1}^{(j)}$ is the decision threshold of the PDBNN speaker model after learning from segment $\mathcal{T}_{n-1}$ at epoch j. We adjusted $\zeta_{n-1}^{(j)}$ whenever misclassification occurs. Specifically, we updated $\zeta_{n-1}^{(j)}$ according to

$$\zeta_n^{(j)} = \begin{cases} \zeta_{n-1}^{(j)} - \eta_r l'(\zeta_{n-1}^{(j)} - S(\mathcal{T}_n)) & \text{if } \mathcal{T}_n \in \mathcal{X}_S \quad \text{and} \quad S(\mathcal{T}_n) < \zeta_{n-1}^{(j)} \\ \zeta_{n-1}^{(j)} + \eta_a l'(S(\mathcal{T}_n) - \zeta_{n-1}^{(j)}) & \text{if } \mathcal{T}_n \in \mathcal{X}_A \quad \text{and} \quad S(\mathcal{T}_n) \geq \zeta_{n-1}^{(j)} \end{cases} \tag{4}$$

where η_r and η_a are respectively the reinforced and anti-reinforced learning parameters (more on next paragraph), $l(d) = \frac{1}{1+e^{-d}}$ is a penalty function, and $l'(d)$ is the derivative of $l(\cdot)$.

In the third modification, we introduced a new method to compute the learning rates. Specifically, the reinforced (anti-reinforced) learning rate η_a (η_r), is proportional to the rate of false rejections (acceptance) weighted by the total number of impostor (speaker) segments:

$$\eta_r = \frac{FRR^{(j-1)}}{FAR^{(j-1)} + FRR^{(j-1)}} \frac{N_{\text{imp}}}{N_{\text{imp}} + N_{\text{spk}}} \eta$$

$$\eta_a = \frac{FAR^{(j-1)}}{FAR^{(j-1)} + FRR^{(j-1)}} \frac{N_{\text{spk}}}{N_{\text{imp}} + N_{\text{spk}}} \eta$$

where $FRR^{(j-1)}$ and $FAR^{(j-1)}$ represent respectively the error rate of false rejections and false acceptances at epoch $j-1$, N_{imp} and N_{spk} represent respectively the total number of training segments from impostors and the registered speaker, and η is a positive learning parameter. This modification aims at increasing the convergence speed of the decision threshold.

6 Pilot Experiments

The architecture of GMMs, EBFNs and PDBNNs depends on several free parameters, including the number of speaker kernels, the number of anti-speaker kernels, and the number of anti-speakers for creating a speaker model. To determine these parameters, a series of pilot experiments involving 30 speakers from the YOHO corpus were performed. Equal error rates (EERs) were used as the performance indicators.

No. of speaker's kernels	EER (%)
10	2.78
20	1.51
40	0.77
80	0.57
160	0.48

(a)

No. of antispeakers	EER (%)
4	2.02
8	1.30
16	0.77
32	0.48
64	0.81

(b)

No. of anti-speaker kernels	EER (%)
40	0.83
80	0.83
160	0.77
320	0.75
640	0.79

(c)

Table 1. Average equal error rates based on 30 GMMs with different numbers of (a) speaker kernels (where the number of anti-speakers and the number of anti-speaker kernels were set to 16 and 160 respectively), (b) anti-speakers (where the number of speaker kernels and anti-speaker kernels were set to 40 and 160 respectively) and (c) anti-speaker kernels (where the number of speaker kernels and anti-speakers were set to 40 and 16 respectively).

Based on the results in Table 1, we used 40 speaker kernels, 160 anti-speaker kernels, and 16 anti-speakers for creating a speaker model in the rest of the experiments. Note that we have selected a sub-optimal number of anti-speakers in order to reduce the computation time in creating the speaker models. As the EBFNs, GMMs and PDBNNs use the same set of kernels, it is not necessary to repeat the above experiments for EBFNs and PDBNNs.

Speaker Model	FAR (%)	FRR (%)	EER (%)
GMMs	8.01	0.08	0.33
EBFs	15.24	0.50	0.48
PDBNNs	1.10	1.87	0.33

Table 2. Average error rates achieved by the GMMs, EBFNs and PDBNNs based on 138 speakers in the YOHO corpus. The pre-defined FAR for GMMs and EBFNs was set to 0.5%.

	PDBNN/GMM		EBFN	
	Train	Test	Train	Test
EER(%)	4.12	24.61	6.86	27.17

Table 3. Performance of the PDBNN, GMM and EBFN in the 2-D speaker classification problem.

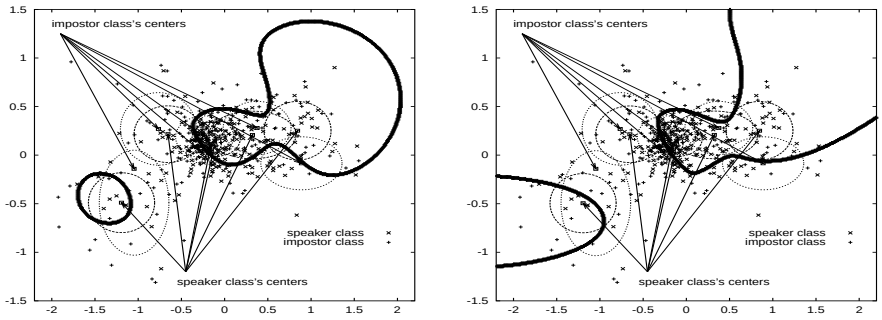
7 Large-Scale Experiments

Table 2 summarizes the average FAR, FRR, and EER obtained by the PDBNN-, GMM- and EBFN-based speaker models. All figures and results were based on the average of 138 speakers in the YOHO corpus. The results, in particular the EER, demonstrate the superiority of the GMMs and PDBNNs over the EBFNs. The EER of GMMs and PDBNNs are the same since their kernel parameters are identical.

In terms of FAR and FRR, Table 2 demonstrates the superiority of the threshold determination procedure of PDBNNs. In particular, Table 2 clearly shows that the globally supervised learning of PDBNNs can make the average FAR very small during verification, whereas the ad hoc approach used by the EBFNs and GMMs is not able to do so. Recall from our previous discussion that the pre-defined FAR was set to 0.5%; however, the average FAR of EBFNs and GMMs are very different from this value.

To illustrate the difference among the PDBNN-, GMM- and EBFN-based speaker models, we extracted the first and second cepstral coefficients of speaker 162 and those of his anti-speakers and impostors to create a set of two-dimensional (2-D) speech data. A PDBNN, a GMM and an EBFN (all with 2 inputs and 6 centers) were trained to classify the patterns into two classes—similar to the enrollment procedure in the speaker verification experiments. Therefore, except for the reduction in feature dimension, the training methods, learning rate and verification methods are identical to the speaker verification experiments described previously.

Table 3 compares the performance of three speaker models, and Figure 1 shows the test data, decision boundaries, function centers, and contours of basis function outputs formed by these models. The decision boundaries are based on the equal error thresholds obtained from the corresponding data set. It is evident from Figure 1(a) that the decision boundaries formed by the EBFN enclose two regions, which belong to the speaker class, with a large amount of test data; whereas, the complement region, which belongs to the impostor class, extends to infinity. On the other hands, the decision boundaries created by the GMM and PDBNN extend to infinity in the feature space for both speaker class and impostor class. Both the decision boundaries (Fig. 1) and the EERs (Table 3) suggest that the GMM and PDBNN provide better generalization than the EBFN. These results also agree with what we have found in Table 2. The poor performance in EBFNs may be caused by the least squares approach to finding the output weights. As the EBFNs formulate the classification problem as a function interpolation problem (mapping from the feature space to 0.0 or 1.0),



(a) (b)

Fig. 1. Speaker classification based on 2-D speech features. The figures plot the decision boundaries, function centres and contours of constant basis function outputs (thin ellipses) produced by (a) EBFNs and (b) GMMs and PDBNNs. Markers 'x' and '+' represent respectively the speaker's data and impostor's data.

overfitting will easily occur if there are too many hidden nodes but too few training samples.

To test the robustness of different speaker models against noise, zero-mean Gaussian noise was added to the YOHO speech so that the resulting corrupted speech has an SNR of 10dB, 6dB, 3dB and 0dB. Tables 4 summarize the average FAR, FRR, and EER obtained by the GMM-, PDBNN- and EBFN-based speaker models under different SNRs. The results show that the error rates of all models increase as the noise power increases. Such performance degradation is mainly caused by the mismatches in training and testing environments.

Evidently, the EERs of PDBNNs and GMMs are smaller than those of EBFNs under different SNRs. Although PDBNNs and GMMs provide better generalization, the performance of PDBNNs and GMMs are still unacceptable at low SNR. In addition to additive noise, telephone speech may also be distorted by the handsets and the telephone channel. We are currently investigating compensation techniques [12] that aim to recover speech signals distorted by both additive and convolutive noise.

	GMM			PDBNN			EBFN		
SNR	FAR	FRR	EER	FAR	FRR	EER	FAR	FRR	EER
0 dB	43.98	55.47	34.00	21.63	76.34	34.00	30.95	66.57	37.51
3 dB	43.52	54.91	27.30	19.52	77.53	27.30	30.48	65.91	30.32
6 dB	42.51	53.59	20.32	17.03	77.53	20.32	29.97	65.16	22.45
10 dB	41.20	50.70	12.79	13.67	76.38	12.79	29.22	61.06	14.58
clean	8.01	0.08	0.33	1.10	1.87	0.33	15.24	0.50	0.48

Table 4. Average error rates (in %) obtained by the GMM, PDBNN and EBFN speaker models at different signal-to-noise ratios.

8 Conclusions

This paper addresses the problem of building a speaker verification system using kernel-based probabilistic neural networks. The modeling capability and robustness of these pattern classifiers are compared. Experimental results, based on 138 speakers and visualization of decision boundaries indicated that GMM- and PDBNN-based speaker models outperform the EBFN ones. Results also show that our modifications on the PDBNN's supervised learning not only makes PDBNNs amenable to speaker verification tasks but also makes their performance more predictable. This work also finds that PDBNNs and GMMs are more robust than EBFNs in recognizing speakers in noisy environments.

References

1. C. Che and Q. Lin. Speaker recognition using HMM with experiments on the YOHO database. In *Eurospeech*, pages 625–628, 1995.
2. F. K. Soong, A. E. Rosenberg, L. R. Rabiner, and B. H. Juang. A vector quantization approach to speaker recognition. In *Proc. ICASSP 85*, pages 387–390, 1985.
3. D. A. Reynolds and R. C. Rose. Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Trans. on Speech and Audio Processing*, 3(1):72–83, 1995.
4. A. Higgins, L. Bahler, and J. Porter. Speaker verification using randomized phrase prompting. *Digital Signal Processing*, 1:89–106, 1991.
5. A. E. Rosenberg, J. Delong, C. H. Lee, B. H. Juang, and F. K. Soong. The use of cohort normalized scores for speaker verification. In *Proc. ICSLP'92*, pages 599–602, 1992.
6. M.W. Mak and S.Y. Kung. Estimation of elliptical basis function parameters by the EM algorithms with application to speaker verification. *IEEE Trans. on Neural Networks*, 11(4):961–969, 2000.
7. K. Farrell, S. Kosonocky, and R. Mammone. Neural tree network/vector quantization probability estimators for speaker recognition. In *Proc. Workshop on Neural Networks for Signal Processing*, pages 279–288, 1994.
8. S. H. Lin, S. Y. Kung, and L. J. Lin. Face recognition/detection by probabilistic decision-based neural network. *IEEE Trans. on Neural Networks, Special Issue on Biometric Identification*, 8(1):114–132, 1997.
9. Jr. J. P. Campbell. Testing with the YOHO CD-ROM voice verification corpus. In *ICASSP'95*, pages 341–344, 1995.
10. C. S. Liu, H. C. Wang, and C. H. Lee. Speaker verification using normalized log-likelihood score. *IEEE Trans on Speech and Audio Processing*, 4(1):56–60, 1996.
11. W. D. Zhang, M. W. Mak, and M. X. He. A two-stage scoring method combining world and cohort model for speaker verification. In *Proc. ICASSP'2000*, June 2000.
12. M. W. Mak and S. Y. Kung. Combining stochastic feautre transformation and handset identification for telephone-based speaker verification. In *Proc. ICASSP'2002*, 2002.