The following publication N. Li, M. Mak and J. Chien, "DNN-Driven Mixture of PLDA for Robust Speaker Verification," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 6, pp. 1371-1383, June 2017 is available at https://doi.org/10.1109/TASLP.2017.2692304.

DNN-driven Mixture of PLDA for Robust Speaker Verification

Na Li, Man-Wai Mak, *Senior Member, IEEE*, and Jen-Tzung Chien, *Senior Member, IEEE*

***Abstract*—The mismatch between enrollment and test utterances due to different types of variabilities is a great challenge in speaker verification. Based on the observation that the SNR-level variability or channel-type variability causes heterogeneous clusters in i-vector space, this paper proposes to apply supervised learning to drive or guide the learning of PLDA mixture models. Specifically, a deep neural network (DNN) is trained to produce the posterior probabilities of different SNR levels or channel types given i-vectors as input. These posteriors then replace the posterior probabilities of indicator variables in the mixture of PLDA. The discriminative training causes the mixture model to perform more reasonable soft divisions of the i-vector space as compared to the conventional mixture of PLDA. During verification, given a test i-vector and a target-speaker's i-vector, the marginal likelihood for the same-speaker hypothesis is obtained by summing the component likelihoods weighted by the component posteriors produced by the DNN, and likewise for the different-speaker hypothesis. Results based on NIST 2012 SRE demonstrate that the proposed scheme leads to better performance under more realistic situations where both training and test utterances cover a wide range of SNRs and different channel types. Unlike the previous SNR-dependent mixture of PLDA which only focuses on SNR mismatch, the proposed model is more general and is potentially applicable to addressing different types of variability in speech.**

***Index Terms*—speaker verification, i-vectors, mixture of PLDA, deep neural networks.**

I. Introduction

MOST state-of-the-art text-independent speaker verification systems use i-vectors as input features. By defining a total variability (TV) space, the posterior means of the latent variables of a factor analyzer [1] are considered as the fixed-length i-vectors of the corresponding utterances. Such a representation greatly simplifies the modeling process as the dimension of i-vectors is much lower than that of GMM-supervectors [2], [3]. However, in addition to speaker information, i-vectors can also be affected by other nuisance variabilities, such as session, channel, and noise-level variabilities commonly found in speech signals. Suppressing the effects caused by these nuisance variabilities but at the same time maintaining speaker information in the i-vectors is a major challenge in i-vector speaker verification.

N. Li and M. W. Mak are with The Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong, SAR (Email: lina011779@126.com; enmwmak@polyu.edu.hk). This work was supported in part by The RGC of Hong Kong SAR (Grant Nos. PolyU 152518/16E and PolyU 152068/15E) and in part by the Taiwan MOST with Grant 105-2221-E-009-137-MY2. J. T. Chien is with the Department of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu 30010, Taiwan (e-mail: jtchien@nctu.edu.tw).

To compensate for session variability, linear discriminant analysis (LDA) [4] followed by within-class covariance normalization (WCCN) [5] are applied to i-vectors; then cosine distance between the target-speaker's i-vector and test i-vector is used as a similarity measure between the target speaker and the test speaker. Recently, probabilistic LDA (PLDA) [6], [7] has become a common backend for i-vector systems. Given a large collection of i-vectors with speaker labels, PLDA shows a powerful data-driven mechanism to separate speaker variability from other undesired variabilities.

While considerable progresses have been made in i-vector extraction, how to develop a noise robust backend classifier remains a major challenge. Although PLDA is very effective at suppressing intersession variability in the i-vector space, it does not consider the effect of noise at varying SNRs. Recent studies [8], [9] suggest that different levels of background noise will shift the i-vectors to different regions of the i-vector space and that i-vectors derived from utterances having similar SNR tend to cluster together. As a result, heterogeneous clusters are formed in the i-vector space.

In view of the above phenomenon, we have previously proposed two approaches for noise robust speaker verification. One is the enhanced SNR-invariant PLDA [8] through which multiple SNR-dependent speaker subspaces are introduced. The other one is SNR-dependent mixture of PLDA [10]. Unlike the conventional mixture of factor analyzers [11] where the posteriors of the indicator variables depend on the data samples, the posteriors of the indicator variables in [10] depend on the SNRs of the utterances. One common characteristic of these two approaches is that the SNR of each test utterance should be estimated when computing the verification score. Although estimating the SNR of an utterance is not difficult,[1] this requirement limits the application of the approaches to handling SNR mismatch only.

Inspired by the clustering phenomenon of i-vectors caused by SNR variability and the success of the DNN/i-vector framework [13], we have recently applied DNNs to guide the training of PLDA mixture models [14]. This paper extends this preliminary work and proposes DNN- and classifier-driven PLDA mixture models to handle both SNR-level variability and session variability in speech signals. Before training the mixture model, training i-vectors are firstly divided into a number of groups (classes) according to the SNR of their utterances. A DNN is then discriminatively trained to produce

[1] SNR estimation will become more difficult when the noise level is high. However, when the waveform of the whole noisy utterance is available, we may use the technique in [12] to increase the contrast between speech and non-speech regions, making the estimated SNR more reliable.

the posterior probabilities of SNR groups given i-vectors as input, which results in an SNR-aware DNN. During the training of the mixture model, the SNR posteriors given by the SNR-aware DNN are used as the posteriors of the indicator variables in the mixture model. The SNR posteriors effectively *drive* the mixture model to capture the cluster properties of i-vectors such that each mixture component (PLDA model) can focus on modeling one SNR-dependent speaker subspace. During the scoring stage, given the i-vectors of the target and test speakers, the SNR posteriors obtained from the DNN are used to linearly combine the marginal likelihoods obtained from different PLDA mixtures. Therefore, unlike the SNR-dependent mixture of PLDA, the actual SNRs of the target and test utterances are not necessary, only their SNR posterior probabilities are needed.

In addition to SNR-level variability, channel-type variability can also cause heterogeneous clusters in i-vector space [15]. We have also investigated the capability of the proposed model in handling channel-type variability in which the training utterances comprise both telephone speech and interview speech but the test utterances comprise interview speech recorded by various microphones. In this case, the DNN is discriminatively trained by using channel types as the class labels. Results based on the core set of NIST 2012 SRE demonstrate the effectiveness of the proposed model for handling two different types of variabilities: SNR-level variability and channel-type variability.

The paper is organized as follows. Section II highlights previous related work on robust i-vector speaker verification. Section III briefly describes the mixture of PLDA. Section IV describes the proposed DNN-driven mixture of PLDA. In Sections V and VI, we report evaluations based on NIST 2012 SRE. Section VII concludes the paper.

II. Related Work

To compensate for the variability of i-vectors caused by different levels of background noise, many strategies have been proposed. One school of thought is to improve the i-vector extraction stage and keep the standard backend unchanged. For example, Hasan and Hansen [16], [17] proposed a two-stage factor analysis scheme in which the posterior means and covariances of acoustic factors are used for computing sufficient statistics in the first stage which are then used for i-vector extraction in the second stage.

Another type of approach is based on multi-condition training. In [18], clean and noisy utterances were pooled together to train a robust PLDA model. In [19], Garcia-Romero *et al* trained multiple PLDA models with tied speaker factors, one for each condition. A robust system was then constructed by combining all of the individual PLDA models according to the posterior probability of each condition. In [20], Villalba and Lleida proposed a multi-channel simplified PLDA; it is a kind of mixture model in which each channel condition (SNR level) is modeled by one channel subspace together with a channel-dependent shift while speaker variability is modeled by a single speaker subspace. The sharing of speaker subspace across all noise conditions requires the assumption that speaker variability are noise-level invariant, which may not be the case in very noisy environments. By assuming that the i-vectors derived from utterances falling within a narrow SNR range should share similar SNR-specific information, Li and Mak [21] proposed to add an SNR-subspace to the conventional PLDA, resulting in SNR-invariant PLDA. In this model, SNR-specific information is separated from speaker-specific information through marginalizing out the SNR factors during the scoring process.

Instead of making PLDA model more amenable to noisy i-vectors, PLDA scores can be robustified by score calibrations where the calibration parameters are dependent on the SNR and duration of target and test utterances [22], [23]. Alternatively, a condition quality vector (q-vector) [24] can be obtained by computing the posterior probabilities of various conditions (formed by combinations of SNR levels and durations). The required offset in score calibration is then a function of the q-vectors corresponding to the target and test utterances, respectively.

In [25], Lei *et al*. proposed extracting i-vectors based on a noise-adapted universal background model (UBM) obtained by adapting the clean UBM to noisy utterances via vector Taylor series (VTS). In [26], [27], i-vectors are denoised in a way similar to spectral subtraction in speech enhancement. However, unlike spectral subtraction, their subtraction model works in the i-vector space. Given an observable noisy i-vector, its clean but unobservable counterpart is approximated by the maximum *a posteriori* (MAP) estimate of the posterior mean of the clean i-vectors. It was recently found that clean i-vectors can be partly restored by translating and rotating the noisy i-vectors, where the translation and rotation matrices are found by the Kabsch algorithm [28]. Instead of denosing the i-vectors, spectral features can also be denoised by DNNs [29] and denoising autoencoder [30] before i-vector extraction.

Due to the excellent performance of deep neural networks (DNN) in many tasks, DNNs have also been applied to speaker verification [13], [31]–[37]. Early work [31], [32] along this direction typically trained a DNN by using acoustic features as input and speaker identities as the target output. Bottleneck features are then extracted from the middle layer of the network. This direct application of DNNs, however, can hardly achieve significant performance gain, although improvement has been found under reverberant environments [32]. A more promising strategy is to incorporate DNNs into i-vector extraction. For example, Lei et al. [13] demonstrated that replacing the standard GMM-based UBM with a phonetically-aware DNN for computing the frame posterior probabilities produces significant performance gain as compared to the standard UBM/i-vector framework. In this DNN/i-vector framework, a phonetically-aware DNN trained for automatic speech recognition (ASR) is used to softly align speech frames to senone categories. Such alignments facilitate the comparison of speakers as if they were pronouncing the same content.

Because the convolution and max-pooling operations in convolution neural networks (CNNs) can reduce the distortion caused by noise, the senone posteriors produced by a CNN have also been used for i-vector extraction [38]. While the

performance of this CNN/i-vector framework is found to be comparable to that of the UBM/i-vector framework, its fusion with the classical i-vector framework produces promising results.

III. Review of PLDA Mixture Models

The PLDA model assumes that the length-normalized i-vectors follow a Gaussian distribution [39]. However, to deal with channel-type variability or SNR variability, the assumption of single Gaussian is rather limited. In such situations, the i-vectors will be better modeled by a mixture of K factor analyzers [7], [40]. This section reviews two kinds of PLDA mixture models: SNR-independent mixture of PLDA and SNR-dependent mixture of PLDA in [10].

A. SNR-Independent Mixture of PLDA

In SNR-independent mixture of PLDA (SI-mPLDA), the posteriors of mixtures are independent of the SNRs of utterances. Essentially, it is a PLDA variant of mixture of factor analyzers in [40]. More precisely, denote $\mathcal{X} = \{\mathbf{x}_{ij}; i = 1,\ldots,S; j = 1,\ldots,H_i\}$ as the set of length-normalized training i-vectors from S speakers, each with H_i utterances. Each i-vector $\mathbf{x}_{ij}$ is assumed to follow a linear weighted sum of K Gaussian densities:

$$\begin{aligned} &p(\mathbf{x}_{ij}) \\ &= \sum\nolimits_{k=1}^{K} \int P(y_{ijk}=1|\mathbf{x}_{ij},\boldsymbol{\theta}) p(\mathbf{x}_{ij}|\mathbf{z}, y_{ijk}=1, \boldsymbol{\theta}_k) p(\mathbf{z}) d\mathbf{z} \\ &= \sum\nolimits_{k=1}^{K} \varphi_k \mathcal{N}(\mathbf{x}_{ij}|\mathbf{m}_k, \mathbf{V}_k\mathbf{V}_k^\top + \boldsymbol{\Sigma}_k), \end{aligned} \tag{1}$$

where y_{ijk} is an indicator variable specifying which of the mixture component is responsible for generating the observation $\mathbf{x}_{ij}$, $\mathbf{z}$ is the speaker factor which is tied across all mixture components, $\boldsymbol{\theta}_k = \{\varphi_k, \mathbf{m}_k, \mathbf{V}_k, \boldsymbol{\Sigma}_k\}$ represent the weight, mean, the speaker subspace, and the covariance matrix of the k-th component, respectively. The parameters of the model in Eq. 1 are denoted as $\boldsymbol{\theta} = \{\boldsymbol{\theta}_k\}_{k=1}^K$.

The EM formulation for estimating the parameters of SNR-independent mixture of PLDA can be derived as follows [10].

- **E-Step**:

$$\langle y_{ijk}|\mathbf{x}_{ij}\rangle = \frac{\varphi_k \mathcal{N}(\mathbf{x}_{ij}|\mathbf{m}_k, \mathbf{V}_k\mathbf{V}_k^\top + \boldsymbol{\Sigma}_k)}{\sum_{k'=1}^{K} \varphi_{k'} \mathcal{N}(\mathbf{x}_{ij}|\mathbf{m}_{k'}, \mathbf{V}_{k'}\mathbf{V}_{k'}^\top + \boldsymbol{\Sigma}_{k'})},$$

$$\mathbf{L}_i = \mathbf{I} + \sum_{k=1}^{K}\sum_{j=1}^{H_i} \langle y_{ijk}|\mathbf{x}_{ij}\rangle \mathbf{V}_k^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{V}_k,$$

$$\langle \mathbf{z}_i|\mathcal{X}\rangle = \mathbf{L}_i^{-1} \sum_{k=1}^{K}\sum_{j=1}^{H_i} \langle y_{ijk}|\mathbf{x}_{ij}\rangle \mathbf{V}_k^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_{ij} - \mathbf{m}_k),$$

$$\langle \mathbf{z}_i\mathbf{z}_i^\top|\mathcal{X}\rangle = \mathbf{L}_i^{-1} + \langle \mathbf{z}_i|\mathcal{X}\rangle \langle \mathbf{z}_i|\mathcal{X}\rangle^\top,$$

- **M-Step**:

$$\mathbf{m}'_k = \frac{\sum_{ij}\langle y_{ijk}|\mathbf{x}_{ij}\rangle \mathbf{x}_{ij}}{\sum_{ij}\langle y_{ijk}|\mathbf{x}_{ij}\rangle}, \qquad \varphi'_k = \frac{\sum_{ij}\langle y_{ijk}|\mathbf{x}_{ij}\rangle}{\sum_{ijl}\langle y_{ijl}|\mathbf{x}_{ij}\rangle},$$

$$\mathbf{V}'_k = \left[\sum_{ij} \langle y_{ijk}|\mathbf{x}_{ij}\rangle \mathbf{f}'_{ijk} \langle \mathbf{z}_i|\mathcal{X}\rangle^\top\right] \left[\sum_i N_{ik} \langle \mathbf{z}_i\mathbf{z}_i^\top|\mathcal{X}\rangle\right]^{-1},$$

$$\boldsymbol{\Sigma}'_k = \frac{\sum_{ij}\left[\langle y_{ijk}|\mathbf{x}_{ij}\rangle \mathbf{f}'_{ijk}\mathbf{f}'^\top_{ijk} - \mathbf{V}'_k \langle \mathbf{z}_i|\mathcal{X}\rangle \langle y_{ijk}|\mathbf{x}_{ij}\rangle \mathbf{f}'^\top_{ijk}\right]}{\sum_i N_{ik}},$$

where

$$\mathbf{f}'_{ijk} = \mathbf{x}_{ij} - \mathbf{m}'_k.$$

Given the target-speaker's i-vector $\mathbf{x}_s$ and a test i-vector $\mathbf{x}_t$, the likelihood ratio $S_{\text{SI-mPLDA}}$ is given by Eq. 2 on next page, where $\hat{\boldsymbol{\Sigma}}_{k_sk_t} = \text{diag}\{\boldsymbol{\Sigma}_{k_s}, \boldsymbol{\Sigma}_{k_t}\}$ and $\hat{\mathbf{V}}_{k_sk_t} = [\mathbf{V}_{k_s}^\top \quad \mathbf{V}_{k_t}^\top]^\top$.

B. SNR-Dependent Mixture of PLDA

Unlike SI-mPLDA, the posteriors of mixtures directly depend on the SNRs of utterances in SNR-dependent mixture of PLDA (SD-mPLDA). In essence, the SNRs are used to guide the clustering process so that each mixture component can focus on one SNR-dependent cluster in the i-vector space. In this model, an i-vector $\mathbf{x}$ is considered generated from the following mixture distribution:

$$\begin{aligned} &p(\mathbf{x}, \ell) = p(\ell) p(\mathbf{x}|\ell) \\ &= p(\ell) \sum\nolimits_{k=1}^{K} \int P(y_k=1|\ell,\lambda) p(\mathbf{x}|\ell, \mathbf{z}, y_k=1, \boldsymbol{\theta}_k) p(\mathbf{z}) d\mathbf{z} \\ &= p(\ell) \sum\nolimits_{k=1}^{K} \gamma_\ell(y_k) \mathcal{N}(\mathbf{x}|\mathbf{m}_k, \mathbf{V}_k\mathbf{V}_k^\top + \boldsymbol{\Sigma}_k), \end{aligned} \tag{3}$$

where ℓ represents the SNR of the utterance whose i-vector is $\mathbf{x}$, y_k is the indicator variable specifying which of the mixture component is responsible for generating $\mathbf{x}$, $\lambda = \{\pi_k, \mu_k, \sigma_k^2\}_{k=1}^K$ denote the parameters of the GMM model trained by using the SNRs of training utterances, $\boldsymbol{\theta}_k = \{\mathbf{m}_k, \mathbf{V}_k, \boldsymbol{\Sigma}_k\}$ represent the mean, the speaker subspace, and the covariance matrix of the k-th component, respectively. The posterior probability of y_k is

$$\gamma_\ell(y_k) \equiv P(y_k=1|\ell,\lambda) = \frac{\pi_k \mathcal{N}(\ell|\mu_k, \sigma_k^2)}{\sum_{k'=1}^{K} \pi_{k'} \mathcal{N}(\ell|\mu_{k'}, \sigma_{k'}^2)}. \tag{4}$$

The EM formulation for estimating the parameters of SNR-independent mixture of PLDA can be derived as follows [10].

- **E-Step**:

$$\langle y_{ijk}|\ell_{ij}\rangle \equiv \gamma_{\ell_{ij}}(y_{ijk}) = \frac{\pi_k \mathcal{N}(\ell_{ij}|\mu_k, \sigma_k^2)}{\sum_{k'=1}^{K} \pi_{k'} \mathcal{N}(\ell_{ij}|\mu_{k'}, \sigma_{k'}^2)},$$

$$\mathbf{L}_i = \mathbf{I} + \sum_{k=1}^{K}\sum_{j=1}^{H_i} \langle y_{ijk}|\ell_{ij}\rangle \mathbf{V}_k^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{V}_k,$$

$$\langle \mathbf{z}_i|\mathcal{X}, \mathcal{L}\rangle = \mathbf{L}_i^{-1} \sum_{k=1}^{K}\sum_{j=1}^{H_i} \langle y_{ijk}|\ell_{ij}\rangle \mathbf{V}_k^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_{ij} - \mathbf{m}_k),$$

$$\langle \mathbf{z}_i\mathbf{z}_i^\top|\mathcal{X}, \mathcal{L}\rangle = \mathbf{L}_i^{-1} + \langle \mathbf{z}_i|\mathcal{X}, \mathcal{L}\rangle \langle \mathbf{z}_i|\mathcal{X}, \mathcal{L}\rangle^\top,$$

- **M-Step**:

$$\mathbf{m}'_k = \frac{\sum_{ij}\langle y_{ijk}|\ell_{ij}\rangle \mathbf{x}_{ij}}{\sum_{ij}\langle y_{ijk}|\ell_{ij}\rangle}, \qquad \pi'_k = \frac{\sum_{ij}\langle y_{ijk}|\ell_{ij}\rangle}{\sum_{ijl}\langle y_{ijl}|\ell_{ij}\rangle},$$

$$\mu'_k = \frac{\sum_{ij}\langle y_{ijk}|\ell_{ij}\rangle \ell_{ij}}{\sum_{ij}\langle y_{ijk}|\ell_{ij}\rangle}; \ \sigma'^2_k = \frac{\sum_{ij}\langle y_{ijk}|\ell_{ij}\rangle (\ell_{ij} - \mu'_k)^2}{\sum_{ij}\langle y_{ijk}|\ell_{ij}\rangle},$$

$$\mathbf{V}'_k = \left[\sum_{ij} \langle y_{ijk}|\ell_{ij}\rangle \mathbf{f}'_{ijk} \langle \mathbf{z}_i|\mathcal{X}, \mathcal{L}\rangle^\top\right] \left[\sum_i N_{ik} \langle \mathbf{z}_i\mathbf{z}_i^\top|\mathcal{X}, \mathcal{L}\rangle\right]^{-1},$$

$$S_{\text{SI-mPLDA}}(\mathbf{x}_s,\mathbf{x}_t)=\ln\frac{\sum_{k_s=1}^{K}\sum_{k_t=1}^{K}\varphi_{k_s}\varphi_{k_t}\mathcal{N}\left(\left[\mathbf{x}_s^\top\ \ \mathbf{x}_t^\top\right]^\top\,|\,\left[\mathbf{m}_{k_s}^\top\ \ \mathbf{m}_{k_t}^\top\right]^\top,\hat{\mathbf{V}}_{k_sk_t}\hat{\mathbf{V}}_{k_sk_t}^\top+\hat{\mathbf{\Sigma}}_{k_sk_t}\right)}{\left[\sum_{k_s=1}^{K}\varphi_{k_s}\mathcal{N}\left(\mathbf{x}_s|\mathbf{m}_{k_s},\mathbf{V}_{k_s}\mathbf{V}_{k_s}^\top+\mathbf{\Sigma}_{k_s}\right)\right]\left[\sum_{k_t=1}^{K}\varphi_{k_t}\mathcal{N}\left(\mathbf{x}_t|\mathbf{m}_{k_t},\mathbf{V}_{k_t}\mathbf{V}_{k_t}^\top+\mathbf{\Sigma}_{k_t}\right)\right]} \tag{2}$$

$$\mathbf{\Sigma}'_k=\frac{\sum_{ij}\left[\langle y_{ijk}|\ell_{ij}\rangle\mathbf{f}'_{ijk}\mathbf{f}'^\top_{ijk}-\mathbf{V}'_k\langle\mathbf{z}_i|\mathcal{X},\mathcal{L}\rangle\langle y_{ijk}|\ell_{ij}\rangle\mathbf{f}'^\top_{ijk}\right]}{\sum_i N_{ik}},$$

where

$$\mathbf{f}'_{ijk}=\mathbf{x}_{ij}-\mathbf{m}'_k.$$

Given the target-speaker's i-vector $\mathbf{x}_s$ and a test i-vector $\mathbf{x}_t$ and the SNR ℓ_s and ℓ_t (in dB) of the corresponding utterances, the likelihood ratio $S_{\text{SD-mPLDA}}$ is given by Eq. 5, where

$$\begin{aligned}\gamma_{\ell_s,\ell_t}(y_{k_s},y_{k_t})&\equiv P(y_{k_s}=1,y_{k_t}=1|\ell_s,\ell_t,\lambda)\\&=\frac{\pi_{k_s}\pi_{k_t}\mathcal{N}([\ell_s\ \ell_t]^\top|[\mu_{k_s}\ \mu_{k_t}]^\top,\text{diag}\{\sigma^2_{k_s},\sigma^2_{k_t}\})}{\sum_{k'_s}\sum_{k'_t}\pi_{k'_s}\pi_{k'_t}\mathcal{N}([\ell_s\ \ell_t]^\top|[\mu_{k'_s}\ \mu_{k'_t}]^\top,\text{diag}\{\sigma^2_{k'_s},\sigma^2_{k'_t}\})}.\end{aligned}$$

Note that a direct implementation of Eq. 2 and Eq. 5 may cause numerical errors. Readers may refer to Appendix A of [10] for a numerical trick to avoid the errors.

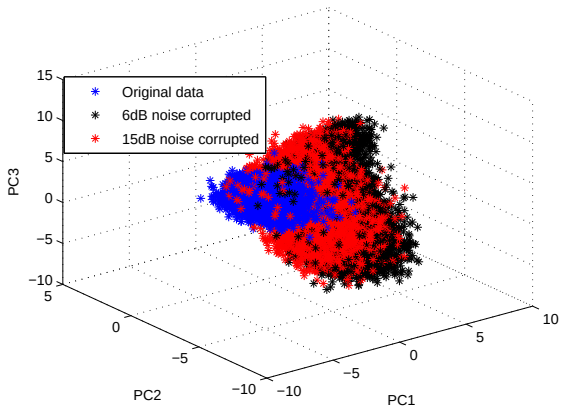


Fig. 1. Illustration of the mean-shift effect of i-vectors (before i-vector preprocessing) caused by different levels of background noise in the corresponding utterances.

IV. DNN-Driven Mixture of PLDA

The work in [15] has demonstrated that when training data exhibit heterogeneous clusters, the conventional mixture of PLDA that performs unsupervised clustering in the feature space outperforms the mixture model in which individual mixture components are trained separately using data from different sources and then combined through the prior probabilities of the sources [41]. This paper aims to introduce the supervisory information to guide this clustering process so that the resulting mixture components become more dependent on their corresponding sources. The sources can be of any form of variabilities in speech, e.g., SNR-level and channel-type. Here, we focus on SNR-level variability and partition the training i-vectors into different groups according to the SNRs of their utterances. Unlike the mixture of PLDA in [41] where a hard decision is made for each training vector as to which cluster it should belong, our proposed method trains multiple mixtures simultaneously and uses soft decisions for aligning the training vectors to the clusters. Unlike the soft decisions in [15], our soft decisions are based on the classifiers that are discriminatively trained to optimally separate the sources. Specifically, SNR-aware DNN was firstly trained using the labeled i-vectors, then, the network outputs (posteriors of SNR subgroups) were used as the posteriors of the indicator variables in the EM algorithm.

A. SNR-aware DNN

The SNR-aware DNN is trained based on the hypothesis that i-vectors extracted from utterances having similar SNRs tend to cluster together, which has been demonstrated in [10]. To show the clustering phenomenon, we added babble noise to 7,156 utterances from NIST 2005–2008 SREs at an SNR of 6dB and 15dB. I-vectors were then extracted from the original (clean) utterances and the noise contaminated utterances. Fig. 1 displays the three groups of i-vectors on the first 3 principal components. Evidently, the i-vectors form three clusters and the locations of the clusters depend on the SNR level. In particular, the 6dB cluster (black) is further away from the clean cluster (blue) than the less noisy cluster (red). Moreover, the cluster shapes are also not identical, meaning that it is better to model the i-vectors by a mixture of PLDA in which each component has its own speaker subspace. To make sure that each component can be estimated by more relevant i-vectors, we divide the training i-vectors into K groups according to the measured SNRs of the utterances, resulting in an SNR level assignment for each training i-vector. The assignments are used as the class labels during DNN training.

The SNR-aware DNN aims to provide supervisory information to assist the clustering of the i-vectors into SNR-dependent groups during the training of the PLDA mixture models. It is believed that a more "crispy" division of the i-vectors can ensure that each mixture can focus on a narrow range of SNRs. To this end, the network should be able to produce the posterior probabilities of SNR levels given i-vectors as input. The network accepts i-vectors as input and produces outputs in an 1-of-K format so that each output node represents one SNR level. The DNN comprises several layers of restricted Boltzmann machines (RBMs) trained by the contrastive divergence algorithm. It is believed that this pre-training step brings the network to a stage that gives better generalization from training data [42]. After pre-training, a softmax output layer is put on the top RBM and the whole network is fine-tuned by the backpropagation algorithm that minimizes the cross-entropy between the desired outputs and actual outputs. After training, the DNN can produce the posterior probabilities of SNR groups given an input i-vector.

$$S_{\text{SD-mPLDA}}(\mathbf{x}_s,\mathbf{x}_t,\ell_s,\ell_t)=\ln\frac{\sum_{k_s=1}^{K}\sum_{k_t=1}^{K}\gamma_{\ell_s,\ell_t}(y_{k_s},y_{k_t})\mathcal{N}\left(\begin{bmatrix}\mathbf{x}_s^\top & \mathbf{x}_t^\top\end{bmatrix}^\top\mid\begin{bmatrix}\mathbf{m}_{k_s}^\top & \mathbf{m}_{k_t}^\top\end{bmatrix}^\top,\hat{\mathbf{V}}_{k_sk_t}\hat{\mathbf{V}}_{k_sk_t}^\top+\hat{\mathbf{\Sigma}}_{k_sk_t}\right)}{\left[\sum_{k_s=1}^{K}\gamma_{\ell_s}(y_{k_s})\mathcal{N}\left(\mathbf{x}_s|\mathbf{m}_{k_s},\mathbf{V}_{k_s}\mathbf{V}_{k_s}^\top+\mathbf{\Sigma}_{k_s}\right)\right]\left[\sum_{k_t=1}^{K}\gamma_{\ell_t}(y_{k_t})\mathcal{N}\left(\mathbf{x}_t|\mathbf{m}_{k_t},\mathbf{V}_{k_t}\mathbf{V}_{k_t}^\top+\mathbf{\Sigma}_{k_t}\right)\right]} \tag{5}$$

B. Generative Model

In the conventional mixture of PLDA (SI-mPLDA in [10]), the training process amounts to unsupervised clustering of the i-vectors into a number of Gaussians, each with a different speaker subspace (PLDA model). Because SNR information is ignored during training, clean i-vectors could be assigned to the mixture component representing noisy i-vectors, and vice versa. To minimize these mis-assignments, we propose to turn the unsupervised clustering to a supervised one by incorporating the SNR information of utterance during model training. More specifically, an SNR-aware DNN is trained according to Section IV-A. For each utterance, an i-vector is extracted and presented to the DNN. The network outputs (posteriors of SNR groups) are then used as the posterior of indicator variables in the mixture model so that the clusters in the i-vector space are more dependent on the SNR levels.

Given a training i-vector $\mathbf{x}_{ij}$ from the *j*-th session of the *i*-th speaker, the posterior probability of the *k*-th SNR group obtained from the SNR-aware DNN is

$$\gamma_{\mathbf{x}_{ij}}(y_{ijk})\equiv P(y_{ijk}=1|\mathbf{x}_{ij},\mathbf{w}), \tag{6}$$

where y_{ijk} is an indicator variable specifying which of the mixture component is responsible for generating the observation $\mathbf{x}_{ij}$ and $\mathbf{w}$ represents the weights of the SNR-aware DNN. With these definitions, the i-vectors are modeled by a mixture of K PLDA models:

$$\begin{aligned}&p(\mathbf{x}_{ij})\\&=\sum_{k=1}^{K}\int P(y_{ijk}=1|\mathbf{x}_{ij},\mathbf{w})p(\mathbf{x}_{ij}|\mathbf{z},y_{ijk}=1,\boldsymbol{\theta}_k)p(\mathbf{z})d\mathbf{z}\\&=\sum_{k=1}^{K}\gamma_{\mathbf{x}_{ij}}(y_{ijk})\mathcal{N}(\mathbf{x}_{ij}|\mathbf{m}_k,\mathbf{V}_k\mathbf{V}_k^\top+\mathbf{\Sigma}_k),\end{aligned} \tag{7}$$

where $\mathbf{z}$ is the speaker factor which is tied across all mixture components, $\mathbf{m}_k$, $\mathbf{V}_k$, and $\mathbf{\Sigma}_k$ represent the mean, the speaker subspace, and the covariance matrix of the *k*-th SNR group, respectively. The parameters of the model in Eq. 7 are denoted as $\boldsymbol{\theta}=\{\mathbf{m}_k,\mathbf{V}_k,\mathbf{\Sigma}_k\}_{k=1}^K$. We assumed that the speaker variability is modeled by $\mathbf{V}_k\mathbf{V}_k^\top$ and that the session variability is modeled by $\mathbf{\Sigma}_k$, where $k=1,\ldots,K$.

C. Model Inference

Denote $\mathcal{Y}=\{y_{ijk}\}_{k=1}^K$ as the set of latent indicator variables specifying which of the K PLDA models should be selected based on the SNRs of training utterances. Specifically, $y_{ijk}=1$ if the *k*-th PLDA model produces $\mathbf{x}_{ij}$, and $y_{ijk}=0$ otherwise. The parameters $\boldsymbol{\theta}$ can be learned from a training set using maximum likelihood estimation. Given an initial value $\boldsymbol{\theta}$, we aim to find a new estimate $\boldsymbol{\theta}'$ that maximizes the following auxiliary function in an EM algorithm:

$$\begin{aligned}&Q(\boldsymbol{\theta}'|\boldsymbol{\theta})=\mathbb{E}_{\mathcal{Y},\mathcal{Z}}\left\{\ln p(\mathcal{X},\mathcal{Y},\mathcal{Z}|\boldsymbol{\theta}')\middle|\mathcal{X},\boldsymbol{\theta}\right\}\\&=\mathbb{E}_{\mathcal{Y},\mathcal{Z}}\left\{\sum\nolimits_{ijk}y_{ijk}\ln\left[p(y_{ijk}=1|\boldsymbol{\theta}')p(\mathbf{x}_{ij}|\mathbf{z}_i,y_{ijk}=1,\boldsymbol{\theta}')p(\mathbf{z}_i)\right]\middle|\mathcal{X},\boldsymbol{\theta}\right\}\end{aligned} \tag{8}$$

where $\mathcal{Z}=\{\mathbf{z}_i;i=1,\ldots,S\}$ is the set of latent variables. To maximize Eq. 8, we need to estimate the posterior expectations of the latent variables given the model parameters $\boldsymbol{\theta}$. The E- and M-steps are as follows:

- **E-Step**:

$$\begin{aligned}\langle y_{ijk}|\mathbf{x}_{ij}\rangle&=\gamma_{\mathbf{x}_{ij}}(y_{ijk}),\\\mathbf{L}_i&=\mathbf{I}+\sum_{k=1}^{K}\sum_{j=1}^{H_i}\langle y_{ijk}|\mathbf{x}_{ij}\rangle\mathbf{V}_k^\top\mathbf{\Sigma}_k^{-1}\mathbf{V}_k,\\\langle\mathbf{z}_i|\mathcal{X}\rangle&=\mathbf{L}_i^{-1}\sum_{k=1}^{K}\sum_{j=1}^{H_i}\langle y_{ijk}|\mathbf{x}_{ij}\rangle\mathbf{V}_k^\top\mathbf{\Sigma}_k^{-1}(\mathbf{x}_{ij}-\mathbf{m}_k),\\\langle\mathbf{z}_i\mathbf{z}_i^\top|\mathcal{X}\rangle&=\mathbf{L}_i^{-1}+\langle\mathbf{z}_i|\mathcal{X}\rangle\langle\mathbf{z}_i|\mathcal{X}\rangle^\top,\end{aligned}$$

where $\gamma_{\mathbf{x}_{ij}}(y_{ijk})$ is obtained from the DNN in Eq. 6.

- **M-Step**:

$$\begin{aligned}\mathbf{m}_k'&=\frac{\sum_{i=1}^{S}\sum_{j=1}^{H_i}\langle y_{ijk}|\mathbf{x}_{ij}\rangle\mathbf{x}_{ij}}{\sum_{i=1}^{S}\sum_{j=1}^{H_i}\langle y_{ijk}|\mathbf{x}_{ij}\rangle},\\\mathbf{V}_k'&=\left\{\sum_{i=1}^{S}\sum_{j=1}^{H_i}\left[\langle y_{ijk}|\mathbf{x}_{ij}\rangle(\mathbf{x}_{ij}-\mathbf{m}_k')\langle\mathbf{z}_i|\mathcal{X}\rangle^\top\right]\right\}\\&\quad\left[\sum_{i=1}^{S}N_{ik}\langle\mathbf{z}_i\mathbf{z}_i^\top|\mathcal{X}\rangle\right]^{-1},\\\mathbf{\Sigma}_k'&=\frac{1}{\sum_i N_{ik}}\sum_{i=1}^{S}\sum_{j=1}^{H_i}\Big[\langle y_{ijk}|\mathbf{x}_{ij}\rangle(\mathbf{x}_{ij}-\mathbf{m}_k')(\mathbf{x}_{ij}-\mathbf{m}_k')^\top\\&\quad-\mathbf{V}_k'\langle\mathbf{z}_i|\mathcal{X}\rangle\langle y_{ijk}|\mathbf{x}_{ij}\rangle(\mathbf{x}_{ij}-\mathbf{m}_k')^\top\Big].\end{aligned}$$

The graphical models of SNR-dependent mixture of PLDA and DNN-driven mixture of PLDA are shown in Fig. 2. We use the underline symbol to represent the set of hyper-parameters of each mixture model. The difference between the two types of mixture models is that the posteriors of the latent indicators in Fig. 2(a) depend on the SNR, whereas the posteriors of the latent indicators in Fig. 2(b) depend on the i-vectors.

D. Likelihood Ratio Score

Given the target-speaker's i-vector $\mathbf{x}_s$ and a test i-vector $\mathbf{x}_t$, and denote $\hat{\mathbf{\Sigma}}_{k_sk_t}=\text{diag}\{\mathbf{\Sigma}_{k_s},\mathbf{\Sigma}_{k_t}\}$, $\hat{\mathbf{V}}_{k_sk_t}=[\mathbf{V}_{k_s}^\top\ \ \mathbf{V}_{k_t}^\top]^\top$. The likelihood ratio $S_{\text{DNN-mPLDA}}$ is shown as Eq. 9 on the next page.

Fig. 3 shows the scoring process in SNR-dependent mixture of PLDA (SD-mPLDA) and DNN-driven mixture of PLDA

$$S_{\text{DNN-mPLDA}}(\mathbf{x}_s, \mathbf{x}_t) = \ln \frac{\sum_{k_s=1}^{K}\sum_{k_t=1}^{K}\gamma_{\mathbf{x}_s}(y_{k_s})\gamma_{\mathbf{x}_t}(y_{k_t})\mathcal{N}\left(\left[\mathbf{x}_s^\top \ \mathbf{x}_t^\top\right]^\top | \left[\mathbf{m}_{k_s}^\top \ \mathbf{m}_{k_t}^\top\right]^\top, \hat{\mathbf{V}}_{k_s k_t}\hat{\mathbf{V}}_{k_s k_t}^\top + \hat{\mathbf{\Sigma}}_{k_s k_t}\right)}{\left[\sum_{k_s=1}^{K}\gamma_{\mathbf{x}_s}(y_{k_s})\mathcal{N}\left(\mathbf{x}_s|\mathbf{m}_{k_s}, \mathbf{V}_{k_s}\mathbf{V}_{k_s}^\top + \mathbf{\Sigma}_{k_s}\right)\right]\left[\sum_{k_t=1}^{K}\gamma_{\mathbf{x}_t}(y_{k_t})\mathcal{N}\left(\mathbf{x}_t|\mathbf{m}_{k_t}, \mathbf{V}_{k_t}\mathbf{V}_{k_t}^\top + \mathbf{\Sigma}_{k_t}\right)\right]} \quad (9)$$

(DNN-mPLDA). The difference is that the former needs to estimate the SNRs of the target and test utterances for obtaining the posteriors of the indicator variables, the latter computes the posteriors of the indicator variables directly given a target i-vector and a test i-vector.

As discussed in [43], given an i-vector pair, the scoring complexity of SD-mPLDA is $\mathcal{O}(K^2D^3)$, where K and D are the number of mixtures and i-vector dimension (after LDA), respectively. Similarly, the scoring complexity of DNN-mPLDA is $\mathcal{O}(K^2(W + D^3))$, where W is the number of weights in the DNN. Because in our experiments $D = 200$ and $W = 120,903$, we have $D^3 \gg W$. This means that the scoring time is mainly spent on computing the Gaussian likelihoods of individual mixtures in Eq. 5 and Eq. 9. As a result, the actual scoring time of SD-mPLDA and DNN-mPLDA is comparable. Note that the actual scoring time also depends on utterance duration, as i-vector extraction has a time complexity proportional to utterance duration.

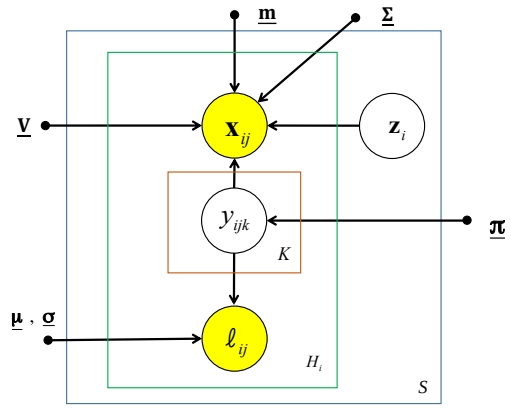
V. Experimental Setup

A. Speech Data and Front-End Processing

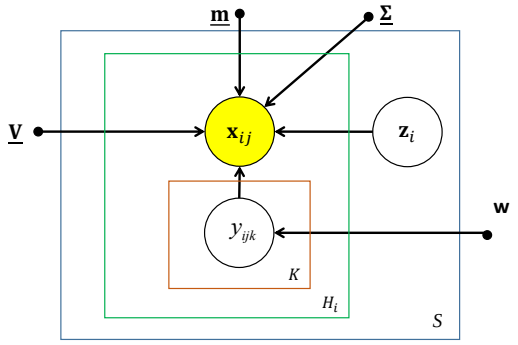
All experiments were performed on the core set of NIST 2012 Speaker Recognition Evaluation (SRE) [44]. We used NIST 2005–2010 data for development and divided the speech data into three categories: (1) development data, (2) test data, and (3) enrollment data.

- *Development Data*: Development data were used for training UBMs, total variability matrices, PLDA models and various PLDA mixture models. More specifically, the microphone and telephone speech files from NIST 2005–2008 SREs were used for training gender-dependent UBMs and total variability matrices. For investigating SNR variability, we added noise to the telephone speech files – excluding speakers with less than two utterances – in NIST 2006–2010 SREs at an SNR of 6dB and 15dB. As a result, for each telephone speech files in these corpora, two noisy speech files were produced. The details of creating the noisy speech files can be found in [21]. The PLDA, PLDA mixture models, and DNNs were trained by the original telephone speech files, noisy telephone speech files and microphone speech files.[2] For investigating channel-type variability, the original telephone speech files and microphone speech files in 2006–2010 SREs, excluding speakers with less than two utterances, were used for training the PLDA, PLDA mixture models and DNNs.
- *Test Data*: All test data were extracted from NIST 2012 SRE, as defined by the `core.ndx` file in the evaluation plan. This paper focuses on common conditions (CC) 1, 3, 4, and 5 of the evaluation plan.
- *Enrollment Data*: For investigating channel-type variability, the enrollment data comprise the conversations of target speakers, as defined by the speaker-table files in NIST 2012 SRE. Each target speaker has one or more conversations recorded over different channels (telephone and microphone) and with different durations. For investigating SNR variability, the enrollment data not only comprise the conversations as defined by the speaker-table files in NIST 2012 SRE but also comprise the noise corrupted telephone conversations of target speakers, at SNRs of 6dB and 15dB. All of the 10-second utterances and summed-channel utterances were removed from the target segments. But we ensured that each target speaker has at least one utterance for enrollment.

[2]Here, microphone speech means telephone conversations recorded over microphone channels in 2008–2010 SRE and interview speech recorded over microphone channels in 2010 SRE.



(a)



(b)

Fig. 2. (a) Probabilistic graphical model representing SNR-dependent mixture of PLDA with parameters $\{\pi_k, \mu_k, \sigma_k, \mathbf{m}_k, \mathbf{V}_k, \mathbf{\Sigma}_k\}_{k=1}^K$. In the diagram, $\underline{\boldsymbol{\pi}} = \{\pi_k\}_{k=1}^K$, $\underline{\boldsymbol{\mu}} = \{\mu_k\}_{k=1}^K$, $\underline{\boldsymbol{\sigma}} = \{\sigma_k\}_{k=1}^K$, $\underline{\mathbf{m}} = \{\mathbf{m}_k\}_{k=1}^K$, $\underline{\mathbf{V}} = \{\mathbf{V}_k\}_{k=1}^K$, $\underline{\mathbf{\Sigma}} = \{\mathbf{\Sigma}_k\}_{k=1}^K$. (b) Probabilistic graphical model representing DNN-driven mixture of PLDA with parameters $\{\mathbf{m}_k, \mathbf{V}_k, \mathbf{\Sigma}_k\}_{k=1}^K$. $\mathbf{w}$ denotes the weights of the SNR-aware DNN.

TABLE I
THE USAGE OF DEVELOPMENT AND TEST DATA FOR DIFFERENT COMMON CONDITIONS IN NIST 2012 SRE. "ORIGINAL": ORIGINAL SPEECH FILES IN 2006–2010 SREs. "TEL": TELEPHONE CONVERSATIONS RECORDED OVER TELEPHONE CHANNELS. "MIC": INTERVIEW SPEECH AND TELEPHONE CONVERSATIONS RECORDED OVER MICROPHONE CHANNELS.

Common Condition	Development Data	Test Trials	
		Enrollment Segments	Test Segments
CC1	Original(tel+mic)	Original(tel+mic)	Interview speech.
CC3	Original(tel+mic)	Original(tel+mic)	Interview speech with added noise.
CC4	Original(tel+mic) + 6dB(tel) + 15dB(tel)	Original(tel) + 6dB(tel) + 15dB(tel)	Telephone speech with added noise.
CC5	Original(tel+mic) + 6dB(tel) + 15dB(tel)	Original(tel) + 6dB(tel) + 15dB(tel)	Telephone speech intentionally collected in a noisy environment.

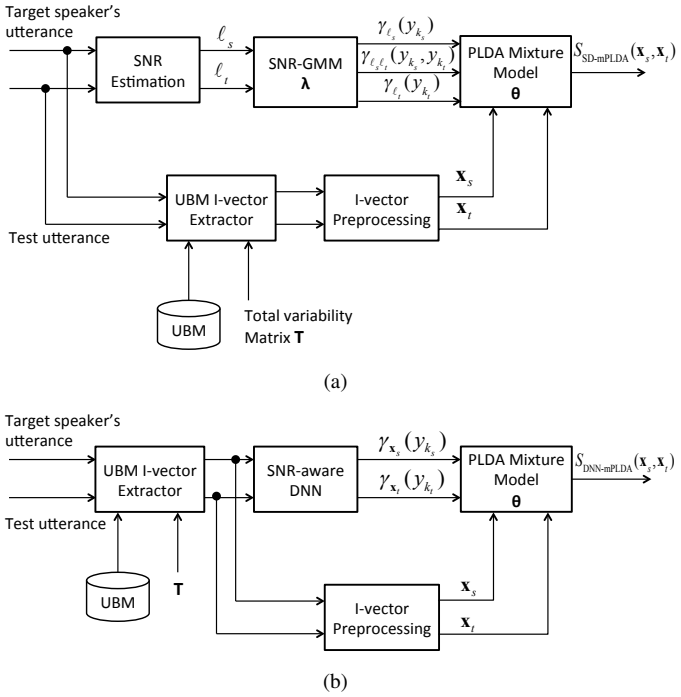


Fig. 3. (a) The scoring process in SNR-dependent mixture of PLDA. ℓ_s and ℓ_t are the SNRs of the target and test utterances, respectively. (b) The scoring process in DNN-driven mixture of PLDA.

Table I shows the usage of the development set and test trials under different common conditions.

A two-channel voice activity detector (VAD) [12], [45] was applied to detect the speech regions of each utterance. 19 Mel frequency cepstral coefficients together with log-energy plus their 1st- and 2nd-derivatives were extracted from the speech regions as detected by the VAD, followed by cepstral mean normalization [46] and feature warping [47] with a window of 3 seconds. A 60-dim acoustic vector was extracted every 10ms, using a Hamming window of 25ms.

B. Variants of PLDA Mixture Models

Because the posteriors of the indicator variables in Eq. 6 can be estimated from other classifiers using i-vectors as input or computed directly from utterances' SNR, our first experiment is to compare the performance of various PLDA mixture models:

- *SI-mPLDA*: SNR-independent mixture of PLDA in [10] where the posteriors $\gamma_{\mathbf{x}_{ij}}(y_{ijk})$'s were obtained from the mixture model itself during training. The prior probability of each mixture component will be used as the mixture weights during scoring. This is similar to the mixture of factor analyzers in [40], except that speaker labels are used in the EM algorithm. In other words, it is a PLDA variant of mixture of factor analyzers.
- *SD-mPLDA*: SNR-dependent mixture of PLDA in [10] where the posterior $\gamma_{\ell_{ij}}(y_{ijk})$, which is obtained from a 1-dimensional (1-D) GMM modeling the SNR ℓ_{ij} distribution, is used to guide the training of mixture of PLDA. During scoring, it is necessary to estimate the SNR of both target and test utterances as they will be used for calculating the posterior probabilities of the indicator variables, which in turn will be used for combining the marginal likelihood arising from different PLDA mixtures.
- *SGMM-mPLDA*: Mixture of PLDA where the posteriors $\gamma_{\mathbf{x}_{ij}}(y_{ijk})$'s were obtained from a supervised GMM using Bayes' theorem. Each of the mixture components was separately trained by using the i-vectors belonging to the same SNR group or channel type. The posteriors were used to guide the training of the mixture models. SNR information is not necessary during scoring.
- *SVM-mPLDA*: Mixture of PLDA where the posteriors $\gamma_{\mathbf{x}_{ij}}(y_{ijk})$'s were obtained from an SVM classifier whose outputs were transformed to posterior probabilities of SNR groups or channel types. Again, the posteriors were used to guide the training of the mixture model. SNR information is not necessary during scoring. The sequential minimal optimization method in [48] was used to train the SVM classifiers with polynomial kernels of degree 3.
- *LR-mPLDA*: Mixture of PLDA where the posteriors $\gamma_{\mathbf{x}_{ij}}(y_{ijk})$'s were obtained from a logistic regression (LR) classifier whose outputs were transformed to posterior probabilities of SNR groups or channel types. SNR information is not necessary during scoring. Iteratively reweighted least squares (IRLS) with the Newton-Raphson algorithm was used for training. The learning rate is 10^{-6} and the maximum of the iteration loops was set to 100.
- *DNN-mPLDA*: The proposed DNN-driven mixture of PLDA where the posteriors $\gamma_{\mathbf{x}_{ij}}(y_{ijk})$'s were used to

guide the training of the mixture model. SNR information is not necessary during scoring. See Section V-D for the details of DNN training.

C. SNR and Channel Variabilities

We conducted two sets of experiments (Exp. A and Exp. B) using different portions of NIST 2012 SRE to demonstrate the effectiveness of the proposed model for handling two different types of variabilities: SNR variability and channel-type variability.

- *Exp. A: SNR-level Variability*. This type of variability occurs when the enrollment utterances and test utterances cover a wide range of SNRs. As a result, during verification, there is a high chance that a clean target-speaker's i-vector (obtained from a clean enrollment utterance) is scored against a noisy test i-vector, and vice versa. To simulate this scenario, we selected Common Conditions (CC) 4 and 5 of NIST 2012 SRE and add babble noise at different SNR levels to the enrollment utterances. Because the test utterances in CC5 were collected in noisy environments and noise has been artificially added to the test utterances in CC4, these two common conditions are ideal for evaluating the capability of different models in tackling SNR variability. In addition to adding babble noise to enrollment utterances, we also added factory noise to the test utterances and have the resulting noisy speech files passing through an artificial reverberator.
- *Exp. B: Channel-type Variability*. This type of variability occurs when the enrollment and test utterances come from different channel types, e.g., telephone speech collected from different handset types or interview sessions recorded by various microphone types. To simulate this scenario, we selected CC1 and CC3 of NIST 2012 SRE. The reason is that in these common conditions, enrollment utterances comprises both telephone speech and interview speech, whereas the test segments comprises interview recordings collected by different microphone types.

For Exp. A, similar to [49], we applied the within-class covariance normalization (WCCN) [5] to whiten the i-vectors, followed by length normalization to reduce the non-Gaussian behavior of the 500-dimensional i-vectors. Then, LDA and WCCN was applied to reduce the intra-speaker variability and emphasize the discriminative information. This procedure reduces the dimension of i-vectors to 200 so that the amount of training data should be sufficient for estimating the PLDA parameters. Then, different types of PLDA models with 150 speaker factors were trained. For Exp. B, as LDA may remove some channel information in the i-vectors, we only applied WCCN after length normalization.

D. Training of DNNs

I-vectors were extracted based on gender-dependent UBMs with 1024 mixtures and total variability matrices with 500 total factors. For Exp. A in Section V-C, the i-vectors for training the SNR-aware DNN were divided into *K* groups according to the measured SNRs of the utterances. The SNRs of the whole training set were divided into K SNR intervals, as shown in Table II. The k-th group comprises the i-vectors whose corresponding utterances have SNR falling in the k-th SNR interval. The DNN comprises 500 Gaussian input nodes and three hidden layers, each having 150 sigmoidal hidden units. One-step contrastive divergence (CD-1) with a mini-batch size of 100 was performed during the pre-training stage. In the fine-tuning stage, conjugate gradient descent was used to minimize the cross-entropy for 30 epochs.

TABLE II
SNR RANGES IN DB FOR DIFFERENT NUMBERS OF SNR GROUPS (K).

K	Group 1	Group 2	Group 3	Group 4	Group 5
2	$(-\infty, 20]$	$(20, \infty)$	–	–	–
3	$(-\infty, 8]$	$(8, 20]$	$(20, \infty)$	–	–
4	$(-\infty, 8]$	$(8, 14]$	$(14, 20]$	$(20, \infty)$	–
5	$(-\infty, 4]$	$(4, 8]$	$(8, 14]$	$(14, 20]$	$(20, \infty)$

For investigating the channel variability (Exp. B), each output node of the channel-aware DNN represents one channel type. Other setups of the channel-aware DNN are the same as those of the SNR-aware DNN.

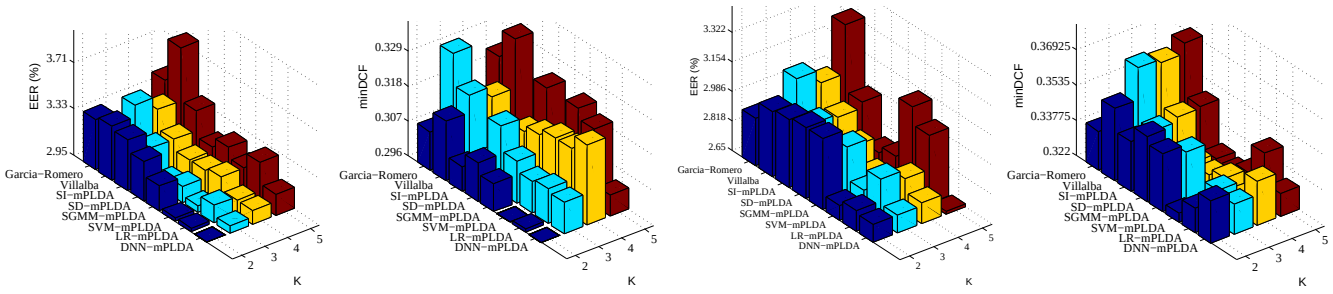
VI. RESULTS AND ANALYSIS

We evaluated the performance of different systems using equal error rate (EER), minimum normalized DCF (minDCF) and actual DCF (actDCF). For minDCF and actDCF, we used the priors and costs according to NIST 2012 SRE [42], i.e., the priors for target speakers were set to 0.01 and 0.001, respectively, and the costs of false acceptance and false rejection were set to 1.0.

A. Performance of Various PLDA Mixture Models

The performance of the multi-condition training method in [19], the multi-channel PLDA in [20] and different PLDA mixture models is shown in Fig. 4. The results show that our proposed classifier-driven PLDA mixture models are better than those in [19] and [20]. Among the proposed classifier-driven models, when the number of mixtures K is smaller than 5, the performance of DNN-mPLDA is comparable to that of SVM-mPLDA. However, DNN-mPLDA performs better than SVM-mPLDA and LR-mPLDA when $K = 5$. This is because we applied the one-vs-rest approach to implement the multi-class SVM and logistic regression classifiers. Increasing the number of classes (e.g., $K = 5$) not only increases the computation complexity of these classifiers, but also reduces their accuracy, leading to higher EER and minDCF. Fig. 4 also shows that SGMM-mPLDA is inferior to DNN-mPLDA, suggesting that supervised GMMs are not capable of classifying the heterogeneous i-vectors. This result also agrees with the finding [13] that the phonetically-aware UBM/i-vector systems cannot outperform the standard UBM/i-vector systems.

Fig 5 shows the DET performance of various mixture models under CC4. For each model, the configuration (by varying K in Fig. 4) that leads to the lowest EER was selected. The results suggest that DNN-mPLDA performs



(a) Male (b) Female

Fig. 4. Performance of Garcia-Romero *et al.* [19], Villalba and Lieida [20], and various PLDA mixture models on CC4 of NIST 2012 SRE core set. To make the performance differences more visible, the vertical axes in these bar charts are not started from zero.

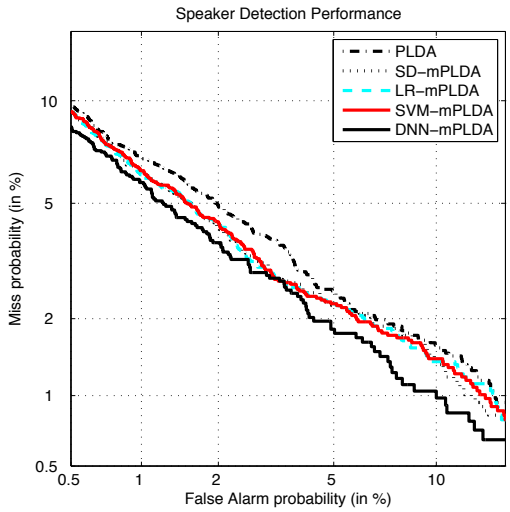


Fig. 5. The DET curves of PLDA and the best performing (in terms of EER) PLDA mixture models in Fig. 4. As the DET curves of SGMM-mPLDA and SVM-mPLDA largely overlap with each other, only SVM-mPLDA is shown. The Rule of 30 for the miss and false alarm probabilities are 1.08% and 0.02%, respectively.

significantly better than other mixture models at most operating points. At the EER points, however, the performance of all mixture models are very close. To verify whether DNN-mPLDA performs better than the others at the EER points, we performed McNemar's tests [50] on the differences between the EERs. The p-values of these tests are shown in Table III. As the p-values between DNN and the other classifiers are less than 0.05, we conclude that DNN-mPLDA outperforms other classifier-driven PLDA mixture models in terms of EER.

Because DNN-mPLDA is generally more stable among other PLDA mixture models when K varies, we will focus on DNN-driven mixture of PLDA in the sequel.

B. Performance on SNR Variability (Exp. A)

This experiment aims to investigate the performance of different models under noise conditions with a wide range of SNRs, and the results are shown in Table IV and Table V. Table IV shows that both SD-mPLDA and DNN-mPLDA outperform the PLDA and SI-mPLDA. In most cases, the proposed DNN-mPLDA performs better than SD-mPLDA. PLDA performs worse than other PLDA mixture models. The reason is that PLDA uses a single model to deal with a wide range of SNRs, whereas the mixture models use a specific mixture component to deal with a much smaller range of SNRs.

In contrast to SI-mPLDA, SNR information is used for assisting the clustering of i-vectors during the training of SD-mPLDA and DNN-mPLDA, which results in more proper i-vector clusters and better SNR-dependent subspace modeling. Another important advantage of SD-mPLDA and DNN-mPLDA is that the verification scores are calculated by combining the PLDA scores using the posterior probabilities of mixture components which are dependent on the test utterances. This leads to a very flexible scoring mechanism. On the other hand, the mixture weights in SI-mPLDA are determined based on the training i-vectors only. Once the weights have been calculated, they will be fixed and used as the priors for the mixture components. As a result, the mixture weights are independent of the test utterances during scoring. As the same combination weights are used regardless of the characteristics of the test utterance, SI-mPLDA is very inflexible.

The main difference between SD-mPLDA and DNN-mPLDA is that the former computes the posteriors of y_{ijk} according to a 1-D GMM that models the SNR distribution and the latter computes the posteriors via an SNR-aware DNN using i-vectors as input. Another difference is that SD-mPLDA relies on SNR information of the test utterances but DNN-mPLDA does not need such information. This trait makes DNN-mPLDA a more general model compared to SD-mPLDA.

Recall from Section V-A and Table II that the training segments comprise the original clean segments and noise contaminated segments with a wide range of SNRs. Our next experiment is to investigate the performance of the proposed model under the situation where the enrollment and development utterances (for PLDA model training) have a wide range of SNRs but the test utterances have a very narrow range of SNRs. To this end, we added different levels of noise to the test segments in CC4 and CC5 of NIST 2012 SRE. The results in Table V show that the proposed model performs better than other models when the SNR distributions of training and test utterances are very different.

It is of interest to see how the mixture models perform when

TABLE III
P-VALUES OF MCNEMAR'S TESTS [50] ON THE DIFFERENCES IN EERS BASED ON CC4 OF NIST 2012 SRE CORE SET, MALE SPEAKERS. FOR EACH MODEL, THE CONFIGURATION (BY VARYING K IN FIG. 4) THAT LEADS TO THE LOWEST EER WAS SELECTED. FOR EACH ENTRY, $p < 0.05$ MEANS THAT THE DIFFERENCE BETWEEN THE EERS IN FIG. 4(A) IS STATISTICALLY SIGNIFICANT AT A CONFIDENCE LEVEL OF 95%.

Method	SGMM-mPLDA	SVM-mPLDA	LR-mPLDA	DNN-mPLDA
SD-mPLDA	0.018	0.479	0.303	0.003
SGMM-mPLDA	–	0.002	0.000	0.000
SVM-mPLDA	–	–	0.669	0.000
LR-mPLDA	–	–	–	0.012

there are noise-type and distortion-type mismatches between the training and test utterances. To this end, we added factory noise from NOISEX-92 to the test utterances, followed by artificial reverberation with $\text{RT}_{60} = 1.044$ seconds. Results in Table VI and Table VII suggest that the proposed model is fairly robust and outperforms other models in most cases even if the test utterances were corrupted by unseen noise type and reverberation effect.

C. Performance on Channel-type Variability (Exp. B)

In addition to SNR variability, channel variability can also cause i-vectors to form clusters. Similar to SNR variability (Exp. A), we trained different classifiers – including supervised GMM (SGMM), SVM, logistic regression, and DNN – to produce the posterior probabilities of telephone and microphone channels given i-vectors as input. Note that SD-mPLDA was excluded from this experiment because it uses auxiliary information (the SNR of utterances) for computing the posterior of indicator variables of the mixture model. For channel-type variability, such information cannot be used.

Table VIII shows the performance of various classifier-driven PLDA mixture models. It suggests that classifier-driven mixture models outperform the baselines (PLDA and SI-mPLDA). The DNN-mPLDA performs slightly better than other models. Judging from the results in Fig. 4 and Table VIII, we conclude that DNNs are the best classifier for PLDA mixture models.

VII. CONCLUSIONS

This paper proposes a new way of applying PLDA mixture models for robust speaker verification. The key idea is to use a classifier to guide the training of PLDA mixture models so that each mixture component precisely models one cluster in the i-vector space. In the testing stage, the verification scores are computed by combining the PLDA scores with dynamic weights depending on the posterior probabilities given by the classifier. The method is flexible in that any discriminatively trained classifiers – including DNN, SVM, and logistic regression – that can leverage the cluster property in the training data can be used. Among them, the DNN classifier was found to achieve the best performance. The proposed method was compared against state-of-the-art models on the NIST SRE 2012 data set. It achieves much better performance than PLDA and conventional mixture of PLDA under SNR-level variability and channel-type variability.

REFERENCES

[1] N. Dehak, R. Dehak, P. Kenny, N. Brümmer, P. Ouellet, and P. Dumouchel, "Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification." in *Proc. Interspeech*, 2009, pp. 1559–1562.

[2] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. ICASSP*, vol. 1, Toulouse, France, May 2006, pp. 97–100.

[3] N. Dehak, R. Dehak, P. Kenny, and P. Dumouchel, "Comparison between factor analysis and GMM support vector machines for speaker verification." in *Proc. of Odyssey*, 2008.

[4] C. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[5] A. Hatch, S. Kajarekar, and A. Stolcke, "Within-class covariance normalization for SVM-based speaker recognition," in *Proc. of the 9th International Conference on Spoken Language Processing*, Pittsburgh, PA, USA, Sep. 2006, pp. 1471–1474.

[6] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. of IEEE 11th International Conference on Computer Vision*, 2007, pp. 1–8.

[7] S. J. Prince, *Computer Vision: Models, Learning, and Inference*. Cambridge University Press, 2012.

[8] N. Li and M. W. Mak, "SNR-invariant PLDA with multiple speaker subspaces," in *Proc. ICASSP*, 2016, pp. 2317–2321.

[9] A. Nautsch, R. Saeidi, C. Rathgeb, and C. Busch, "Analysis of mutual duration and noise effects in speaker recognition: benefits of condition-matched cohort selection in score normalization." in *Proc. Interspeech*, 2015, pp. 3006–3010.

[10] M. W. Mak, X. M. Pang, and J. T. Chien, "Mixture of PLDA for noise robust i-vector speaker verification," *IEEE/ACM Trans. on Audio, Speech and Language Processing*, vol. 24, no. 1, pp. 130–142, 2016.

[11] G. McLachlan and D. Peel, "Mixtures of factor analyzers," *New York, NY, USA: Wiley, Finite Mixture Models*, pp. 238–256, 2000.

[12] M. W. Mak and H. B. Yu, "A study of voice activity detection techniques for NIST speaker recognition evaluations," *Computer, Speech and Language*, vol. 28, no. 1, pp. 295–313, Jan 2014.

[13] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proc. ICASSP*, 2014, pp. 1695–1699.

[14] N. Li, M. W. Mak, and T. J. Chien, "Deep neural network driven mixture of PLDA for robust i-vector speaker verification," in *Proc. IEEE SLT 2016 Workshop on Spoken Language Technology*. San Diego: IEEE, Dec. 2016.

[15] T. Pekhovsky and A. Sizov, "Comparison between supervised and unsupervised learning of probabilistic linear discriminant analysis mixture models for speaker verification," *Pattern Recognition Letters*, vol. 34, no. 11, pp. 1307–1313, 2013.

[16] T. Hasan and J. Hansen, "Acoustic factor analysis for robust speaker verification," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 842–853, 2013.

[17] ——, "Maximum likelihood acoustic factor analysis models for robust speaker verification in noise," *IEEE Transactions on Audio, Speech, And Language Processing*, vol. 22, no. 2, pp. 381–391, 2014.

[18] T. Hasan, S. O. Sadjadi, G. Liu, N. Shokouhi, H. Boril, and J. H. Hansen, "CRSS systems for 2012 NIST speaker recognition evaluation," in *Proc. ICASSP*, 2013, pp. 6783–6787.

[19] D. Garcia-Romero, X. Zhou, and C. Espy-Wilson, "Multicondition training of Gaussian PLDA models in i-vector space for noise and

TABLE IV
Exp. A: PERFORMANCE OF PLDA, SI-MPLDA, SD-MPLDA, AND DNN-MPLDA ON CC4 AND CC5 OF NIST 2012 SRE CORE SET. K IS THE NUMBER OF MIXTURES IN THE MIXTURE MODELS.

Method	K	Male						Female					
		CC4			CC5			CC4			CC5		
		EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
PLDA	–	3.49	0.308	0.426	2.97	0.290	0.368	3.14	0.353	0.443	2.47	0.346	0.398
SI-mPLDA	2	3.49	0.303	0.418	3.04	0.300	0.373	3.11	0.350	0.440	2.55	0.340	0.380
	3	3.31	0.302	0.420	3.06	0.286	0.371	3.02	0.351	0.437	2.41	0.345	0.377
	4	3.31	0.299	0.421	2.93	0.288	0.379	3.00	0.354	0.425	2.60	0.332	0.377
	5	3.52	0.301	0.416	3.48	0.303	0.378	3.04	0.355	0.446	2.71	0.355	0.396
SD-mPLDA	2	3.37	0.307	0.422	2.92	0.298	0.363	3.13	0.359	0.456	2.50	0.344	0.375
	3	3.06	0.315	0.411	2.80	**0.276**	0.360	**2.65**	**0.331**	0.410	2.38	0.324	0.380
	4	3.20	0.311	0.418	2.87	0.284	0.365	2.88	0.334	0.431	2.38	0.347	0.394
	5	3.24	0.321	0.419	2.87	0.287	0.368	2.77	**0.331**	0.419	2.46	0.332	0.379
DNN-mPLDA	2	**2.95**	**0.296**	**0.409**	2.86	0.282	0.367	2.77	0.346	0.428	2.38	0.326	0.371
	3	3.03	0.305	0.418	**2.73**	0.279	0.366	2.77	0.339	**0.403**	**2.36**	0.333	**0.364**
	4	3.10	0.319	0.420	2.78	0.278	0.365	2.79	0.347	0.420	2.38	0.329	0.377
	5	3.18	0.302	0.413	2.87	0.278	**0.356**	2.67	0.335	0.441	2.51	**0.323**	0.394

TABLE V
Exp. A: PERFORMANCE OF PLDA, SI-MPLDA, SD-MPLDA, AND DNN-MPLDA ON CC4 AND CC5 OF NIST 2012 SRE CORE SET, FEMALE SPEAKERS. BABBLE NOISE WAS ADDED TO THE TEST SEGMENTS AT DIFFERENT LEVELS OF SNR. K WAS SET TO 3 FOR THE MIXTURE MODELS.

Method	CC4 (6dB)			CC4 (15dB)			CC5 (6dB)			CC5 (15dB)		
	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
PLDA	3.16	0.403	0.474	2.81	0.360	0.441	6.05	0.589	0.648	3.48	0.410	0.468
SI-mPLDA	3.14	0.396	0.457	3.00	0.360	0.433	5.96	0.572	0.629	3.49	0.402	0.476
SD-mPLDA	3.07	0.412	0.468	2.62	**0.349**	0.429	**5.86**	**0.560**	0.627	3.31	**0.385**	0.463
DNN-mPLDA	**2.97**	**0.379**	**0.454**	**2.58**	**0.349**	**0.416**	5.87	0.565	**0.626**	**3.28**	0.389	**0.452**

TABLE VI
Exp. A: PERFORMANCE OF PLDA, SI-MPLDA, SD-MPLDA, AND DNN-MPLDA ON CC4 AND CC5 OF NIST 2012 SRE CORE SET, FEMALE SPEAKERS. FACTORY NOISE WAS ADDED TO THE TEST SEGMENTS AT DIFFERENT LEVELS OF SNR. THE SAME MIXTURE MODELS AS IN TABLE V WERE USED FOR SCORING.

Method	CC4 (6dB)			CC4 (15dB)			CC5 (6dB)			CC5 (15dB)		
	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
PLDA	5.60	**0.469**	0.659	3.27	0.370	0.485	5.93	0.557	0.646	3.25	0.417	0.465
SI-mPLDA	5.41	0.492	0.646	3.27	0.365	0.480	5.62	0.549	0.635	3.20	0.389	0.470
SD-mPLDA	5.50	0.519	0.656	3.22	0.363	0.478	5.29	**0.536**	0.635	**3.08**	0.390	0.461
DNN-mPLDA	**5.33**	0.475	**0.645**	**3.03**	**0.359**	**0.476**	**5.25**	0.544	**0.629**	3.10	**0.376**	**0.460**

reverberation robust speaker recognition," in *Proc. ICASSP*, 2012, pp. 4257–4260.

[20] J. Villalba and E. Lleida, "Handling i-vectors from different recording conditions using multi-channel simplified PLDA in speaker recognition." in *Proc. ICASSP*, 2013, pp. 6763–6767.

[21] N. Li and M. W. Mak, "SNR-invariant PLDA modeling in nonparametric subspace for robust speaker verification," *IEEE/ACM Trans. on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1648–1659, 2015.

[22] M. I. Mandasari, R. Saeidi, M. McLaren, and D. A. van Leeuwen, "Quality measure functions for calibration of speaker recognition systems in various duration conditions," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 11, pp. 2425–2438, 2013.

[23] M. I. Mandasari, R. Saeidi, and D. A. van Leeuwen, "Quality measures based calibration with duration and noise dependency for speaker recognition," *Speech Communication*, vol. 72, pp. 126–137, 2015.

[24] A. Nautsch, R. Saeidi, C. Rathgeb, and C. Busch, "Robustness of quality-based score calibration of speaker recognition systems with respect to low-SNR and short-duration conditions," in *Proc. Odyssey*, 2016, pp. 358–365.

[25] Y. Lei, L. Burget, and N. Scheffer, "A noise robust i-vector extractor using vector Taylor series for speaker recognition," in *Proc. ICASSP*, 2013, pp. 6788–6791.

[26] W. B. Kheder, D. Matrouf, J.-F. Bonastre, M. Ajili, and P.-M. Bousquet, "Additive noise compensation in the i-vector space for speaker recognition," in *Proc. ICASSP*, 2015, pp. 4190–4194.

[27] Y. Solewicz, H. Aronowitz, and T. Becker, "Reducing noise bias in the i-vector space for speaker recognition," in *Odyssey 2016: The Speaker and Language Recognition Workshop*, Bilbao, Spain, June 21-24 2016, pp. 372–376.

[28] W. B. Kheder, D. Matrouf, M. Ajili, and J.-F. Bonastre, "Iterative Bayesian and MMSE-based noise compensation techniques for speaker recognition in the i-vector space," in *Proc. Odyssey*, 2016, pp. 60–67.

[29] F. Richardson, B. Nemsick, and D. Reynolds, "Channel compensation for speaker recognition using MAP adapted PLDA and denoising

TABLE VII
PERFORMANCE OF PLDA, SI-MPLDA, SD-MPLDA, AND DNN-MPLDA ON CC4 AND CC5 OF NIST 2012 SRE CORE SET, FEMALE SPEAKERS. TEST SEGMENTS WERE CONTAMINATED BY FACTORY NOISE AND REVERBERATION EFFECT. THE SAME MIXTURE MODELS AS IN TABLE V WERE USED FOR SCORING.

Method	CC4 (6dB)			CC4 (15dB)			CC5 (6dB)			CC5 (15dB)		
	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
PLDA	14.90	0.972	0.979	10.47	0.911	0.922	17.39	0.944	0.985	12.08	0.883	0.928
SI-mPLDA	14.26	**0.956**	0.974	9.96	0.896	0.922	16.52	**0.933**	0.982	11.94	0.867	0.919
SD-mPLDA	14.05	0.959	0.974	9.94	0.885	**0.905**	16.14	0.952	0.978	11.90	0.873	0.918
DNN-mPLDA	**14.02**	0.957	**0.973**	**9.67**	**0.872**	0.909	**15.94**	0.942	**0.976**	**11.78**	**0.840**	**0.911**

TABLE VIII
Exp. B: PERFORMANCE OF PLDA, SI-MPLDA, AND DIFFERENT TYPES OF CLASSIFIER-DRIVEN MPLDA ON CC1 AND CC3 OF NIST 2012 SRE CORE SET, MALE SPEAKERS. THE NUMBER OF MIXTURES (K) WAS SET TO 2.

Method	CC1			CC3		
	EER(%)	minDCF	actDCF	EER(%)	minDCF	actDCF
PLDA	6.25	0.378	0.534	5.88	0.278	0.609
SI-mPLDA	6.40	0.381	0.548	5.83	0.276	0.603
SGMM-mPLDA	**5.92**	0.380	0.534	5.52	**0.272**	0.592
SVM-mPLDA	6.12	0.379	0.527	5.59	0.274	0.606
LR-mPLDA	6.14	0.378	0.524	5.62	0.278	0.603
DNN-mPLDA	5.98	**0.370**	**0.514**	**5.51**	0.274	**0.578**

DNNs," in *Proc. Odyssey*, 2016, pp. 225–230.
[30] Z. Tan, Y. Zhu, M. W. Mak, and B. Mak, "Senone i-vectors for robust speaker verification," in *Proc. of Int. Sym. on Chinese Spoken Language Processing (ISCSLP)*, Tianjin, China, October 2016.
[31] S. Yaman, J. W. Pelecanos, and R. Sarikaya, "Bottleneck features for speaker recognition." in *Proc. Odyssey*, 2012, pp. 105–108.
[32] T. Yamada, L. Wang, and A. Kai, "Improvement of distant-talking speaker identification using bottleneck features of DNN." in *Proc. Interspeech*, 2013, pp. 3661–3664.
[33] O. Ghahabi and J. Hernando, "Deep belief networks for i-vector based speaker recognition," in *Proc. ICASSP*, 2014, pp. 1700–1704.
[34] E. Variani, X. Lei, E. McDermott, I. Lopez Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. ICASSP*, 2014, pp. 4052–4056.
[35] Y. Tian, M. Cai, L. He, and J. Liu, "Investigation of bottleneck features and multilingual deep neural networks for speaker verification," in *Proc. Interspeech*, 2015, pp. 1151–1155.
[36] D. Garcia-Romero and A. McCree, "Insights into deep neural networks for speaker recognition," in *Proc. Interspeech*, 2015, pp. 1141–1145.
[37] X. Zhao, Y. Wang, and D. Wang, "Deep neural networks for cochannel speaker identification," in *Proc. ICASSP*, 2015, pp. 4824–4828.
[38] M. McLaren, Y. Lei, N. Scheffer, and L. Ferrer, "Application of convolutional neural networks to speaker recognition in noisy conditions." in *Proc. Interspeech*, 2014, pp. 686–690.
[39] D. Garcia-Romero and C. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Proc. Interspeech*, 2011, pp. 249–252.
[40] Z. Ghahramani and G. Hinton, "The EM algorithm for mixtures of factor analyzers," Dept. of Comput. Sci, University of Toronto, Canada, CRG-TR-96-1, Tech. Rep., 1996.
[41] M. Senoussaoui, P. Kenny, N. Brümmer, E. De Villiers, and P. Dumouchel, "Mixture of PLDA models in i-vector space for gender-independent speaker recognition," in *Proc. Interspeech*, 2011, pp. 25–28.
[42] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?" *Journal of Machine Learning Research*, vol. 11, no. Feb, pp. 625–660, 2010.
[43] M. W. Mak, "Fast scoring for mixture of PLDA in i-vector/plda speaker verification," in *Proc. Asia-Pacific Signal and Information Processing Association, Annual Summit and Conference (APSIPA ASC)*, Hong Kong, Dec. 2015, pp. 587–593.
[44] NIST, "The NIST year 2012 speaker recognition evaluation plan," *http://www.nist.gov/itl/iad/mig/sre12.cfm*, 2012.
[45] H. Yu and M. Mak, "Comparison of voice activity detectors for interview speech in NIST speaker recognition evaluation," in *Proc. of Interspeech*, 2011, pp. 2353–2356.
[46] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *J. Acoust. Soc. Am.*, vol. 55, no. 6, pp. 1304–1312, Jun. 1974.
[47] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, Crete, Greece, Jun. 2001, pp. 213–218.
[48] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.
[49] M. McLaren, M. Mandasari, and D. Leeuwen, "Source normalization for language-independent speaker recognition using i-vectors," in *Odyssey 2012: The Speaker and Language Recognition Workshop*, 2012, pp. 55–61.
[50] L. Gillick and S. J. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *Proc. ICASSP*, vol. 1, Glasgow, UK, May 1989, pp. 532–535.

Na Li received the B.S. degree in Environmental Engineering, M.S. and Ph.D. degrees in Acoustics from Northwestern Polytechnic University (NPU), Xi'an, China, in 2007, 2010, and 2015, respectively. From 2011 to 2013, she served as a Research Assistant in Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China. Between 2014 and 2015, she was a Research Associate in the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, and become a PostDoc Fellow since 2016. She is currently Senior Researcher of Tencent AI Lab, China. Dr Li's research interests include speaker recognition, voice conversion, and machine learning.

Man-Wai Mak (M'93–SM'15) received a PhD in Electronic Engineering from the University of Northumbria in 1993. He joined the Department of Electronic and Information Engineering at The Hong Kong Polytechnic University in 1993 and is currently an Associate Professor in the same department. He has authored more than 170 technical articles in speaker recognition, machine learning, and bioinformatics. Dr. Mak also coauthored a postgraduate textbook *Biometric Authentication: A Machine Learning Approach, Prentice Hall, 2005* and a research monograph *Machine Learning for Protein Subcellular Localization Prediction, De Gruyter, 2015*. He served as a member of the IEEE Machine Learning for Signal Processing Technical Committee in 2005-2007. He served as an associate editor of IEEE/ACM Transactions on Audio, Speech and Language Processing in 2011–2014. He is currently an associate editor of Journal of Signal Processing Systems, Advances in Artificial Neural Systems, and IEEE Biometrics Compendium. Dr Mak gave a tutorial on machine learning for speaker recognition in Interspeech'2016. He also served as Technical Committee Members of a number of international conferences, including ICASSP and Interspeech. Dr. Mak's research interests include speaker recognition, machine learning, and bioinformatics.

Jen-Tzung Chien (M'97-SM'04) received his Ph.D. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan in 1997. He is now with the Department of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu, where he is currently a University Chair Professor. His research interests include machine learning, deep learning, speaker recognition, and face recognition. Dr. Chien served as the associate editor of the IEEE Signal Processing Letters in 2008-2011, the guest editor of the IEEE Transactions on Audio, Speech, and Language Processing in 2012, the tutorial speaker of the Interspeech in 2013 and 2016 and the ICASSP in 2012, 2015 and 2017. He received the Best Paper Award of the IEEE Automatic Speech Recognition and Understanding Workshop in 2011. He has published extensively, including the book "Bayesian Speech and Language Processing", Cambridge University Press, 2015. He currently serves as an elected member of the IEEE Machine Learning for Signal Processing Technical Committee. He is the General Co-Chair of the IEEE International Workshop on Machine Learning for Signal Processing in 2017.