CONTENT-AWARE LOCAL VARIABILITY VECTOR FOR SPEAKER VERIFICATION WITH SHORT UTTERANCE

Liping Chen[1,2,3]*, Kong Aik Lee*[2]*, Eng-Siong Chng*[3]*, Bin Ma*[2]*, Haizhou Li*[2]*, and Li Rong Dai*[1]

[1]National Engineering Laboratory for Speech and Language Information Processing, USTC, China
[2]Institute for Infocomm Research, A*STAR, Singapore
[3]Temasek Laboratories, NTU, Singapore

clp2011@mail.ustc.edu.cn kalee@i2r.a-star.edu.sg

ABSTRACT

I-vector has shown to be very effective in speaker verification with long-duration speech utterances. But when test utterances are of short duration, content mismatch between the enrollment and test utterances limit the performance of i-vector system. This paper proposes to extract local session variability vectors on different phonetic classes from the utterances instead of estimating the session variability across the whole utterance as i-vector does. Using the posteriors given by a deep neural network (DNN) trained for phone state classification, the local vectors represent the session variability contained in specific phonetic content. Our experiments show that the content-aware local vectors are better at coping with the content mismatch between training and test utterances of short durations for text-independent, text-constrained and text-dependent tasks.

Index Terms— content-aware local variability, short-duration utterance, speaker verification

1. INTRODUCTION

Over recent years, many approaches based on the *Gaussian mixture model-universal background model* (GMM-UBM) [1] have been proposed for speaker verification [2], among which i-vector has now become the mainstream method [3]. Similar to a GMM supervector [4], an i-vector is a fixed-length representation of a speech utterance, which typically consists of varying number of frames. Different from the supervector, an i-vector has a much lower dimensionality. Channel compensation could therefore be performed using the *probabilistic linear discriminant analysis* (PLDA) [5], which also serves as the backend classifier. In [6, 7], a deep neural network (DNN) trained for phone state classification was used to assume the role of GMM, where frame alignment is now given by the phone state posterior. The strong discriminative property of DNN improves the frame alignment, and therefore the speaker verification performance.

I-vectors have proven to be very effective for text-independent speaker verification when both the training and test utterances are of long duration. For short duration, the content mismatch between the training and test utterances makes it difficult for i-vectors to be applied directly [8]. In [9, 10, 11], we proposed to model the local session variability with respect to the Gaussian components or dimensions of acoustic features. The local vectors offer a flexible way to compare speaker information by matching specific phonetic context in a much similar fashion as in forensic speaker recognition [12]. However, the phonetic contents estimated with the local vectors were ambiguous since the GMM was trained in an unsupervised way and the dimensions of the acoustic feature vectors do not have definite phonetic interpretations. In [13], frame alignment by a DNN trained for monophonic phone-state classification was used to estimate phone-centric local vectors for the monophones. In this way, each local vector is endowed with clear phonetic interpretation. By implementing the PLDA scoring in a phone-aware manner, the local vectors achieved performance improvement with respect to i-vectors, providing a solution to the content matching for short-duration utterances.

In [13], the experiments were conducted on a text-constrained task. In this paper, we extend the scope of content matching to a more general text-independent task and resort to the posterior of a DNN acoustic model trained on tied-triphone Markov states (senones). We propose to cluster the senones with each cluster representing the phonetic information of similar senones by treating the senones as Gaussian-like units. A local variability vector is extracted for each cluster as an explicit representation of the session variability contained in the cluster, called the content-aware local vector. A PLDA model is trained on the local vectors for channel compensation. In the PLDA modeling, the local vectors are modeled in a content-aware manner but with the correlations among them being considered.

The rest of this paper is organized as follows. Section 2 gives an introduction to DNN i-vector. The content-aware local variability vector is presented in Section 3. In Section 4, the content-aware PLDA is described. The experiments are given in Section 5. We draw our conclusions in Section 6.

2. DNN I-VECTOR

2.1. I-vector framework

The fundamental assumption of i-vector is that the feature vector sequence of the utterance is generated from a session-specific GMM. Furthermore, the mean supervector of the GMM $\mathbf{m}_r$ is confined to a low-dimensional subspace $\mathbf{T}$ with origin μ, as follows

$$\mathbf{m}_r = \mu + \mathbf{T}\mathbf{w}_r \tag{1}$$

where $r = 1, 2, ..., R$ indexes the sessions. The matrix $\mathbf{T}$ is referred to as the total variability matrix, and the model as the total variability model. The latent variable $\mathbf{w}_r$ is session-specific whose posterior mean is the so-called i-vector [3].

This work of Liping Chen is partly supported by the National Nature Science Foundation of China (Grant No. 61273264), Science and Technology Department of Anhui Province (Grant No. 15CZZ02007) and the Chinese Academy of Sciences (Grant No. XDB02070006)

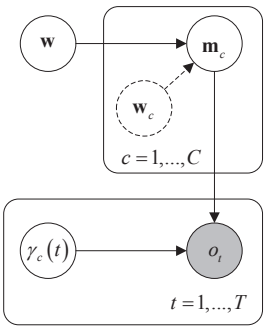

Fig. 1. Graphical model illustrating the difference between the total and local variability models. The latent variable $\mathbf{w}$ is used for total variability model, while $\mathbf{w}_c$, for $c = 1, 2, ..., C$, are for local variability model. For brevity, the session index r is dropped .

Fig. 1 shows the graphical model of the total variability model. The subscript $c = 1, ..., C$ are the indices to the C Gaussian components in the model; and $\mathbf{m}_c$ is the mean vector of the c-th Gaussian. Given a speech utterance $\mathcal{O} = \{o_1, ..., o_T\}$ with T to be the number of frames, $\gamma_c(t)$ is the occupancy of the t-th frame o_t to the c-th Gaussian. The channel and speaker variability observed in all the Gaussian components are modeled jointly with the latent variable $\mathbf{w}$. This is reflected by allocating $\mathbf{w}$ outside the rectangular box for the Gaussians with $c = 1, ..., C$ at the bottom.

2.2. DNN i-vector

In [6] and [7], DNN trained for phone state classification was used to align the frames. In the state-of-the-art automatic *speech recognition*(ASR) systems, context-dependent phones (e.g., triphones) are represented by a number of hidden Markov states. Typically, the Markov states are tied across the context-dependent phones using a decision tree trained with a maximum likelihood (ML) criterion [14]. In the GMM-HMM framework, the emission probabilities of the tied-states are modeled with GMMs [15], while in a DNN-HMM hybrid system, with a DNN. More precisely, each output node of the DNN is trained to estimate the posterior probability of tied-states given the acoustic observations [16].

The DNN for senonic acoustic modeling is illustrated in Fig. 2. A set of C tied-states, i.e., senones, denoted as $\mathcal{S} = \{s_1, s_2, ..., s_C\}$ is modeled at the output layer. Given an input observation $\mathbf{x}_t$ (usually concatenated with contextual frames), the values of the output nodes are the senonic posterior probability $p(s_c|\mathbf{x}_t)$. By treating the set of senones $\mathcal{S}$ modeled by the DNN as Gaussian-like units in the UBM, it was shown in [6] and [7] that the Baum-Welch statistics, and therefore the i-vector, could be extracted by replacing the frame alignment with the senone posterior:

$$\gamma_c(t) \leftarrow p(s_c|\mathbf{x}_t) \tag{2}$$

Using the frame occupancy $\gamma_c(t)$, the mean vector and covariance matrix of the c-th Gaussian in the total variability model are computed as follows:

$$\mu_c = \frac{\sum_t \gamma_c(t)\, o_t}{\sum_t \gamma_c(t)} \tag{3}$$

$$\mathbf{\Sigma}_c = \frac{\sum_t \gamma_c(t)\,(o_t - \mu_c)\,(o_t - \mu_c)^{\mathsf{T}}}{\sum_t \gamma_c(t)} \tag{4}$$

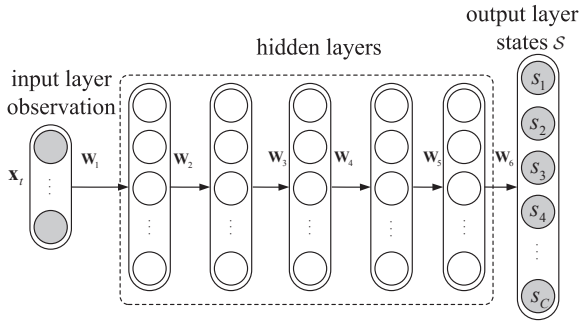

Fig. 2. A deep neural network (DNN) trained for phone state classification. Here, $\mathcal{W} = \{\mathbf{W}_1, ..., \mathbf{W}_6\}$ is a set of the network parameters (weight matrices and biases)).

The DNN i-vector is estimated as:

$$\phi = \left[\mathbf{I} + \sum_{c=1}^{C} N_c \mathbf{T}_c^{\mathsf{T}} \mathbf{T}_c\right]^{-1} \left[\sum_{c=1}^{C} \mathbf{T}_c^{\mathsf{T}} \widetilde{\mathbf{F}}_c\right] \tag{5}$$

where N_c is the zero-order statistics computed as

$$N_c = \sum\nolimits_t \gamma_c(t) \tag{6}$$

and $\widetilde{\mathbf{F}}_c$ is the first-order statistics centered to the mean μ_c and whitened with respect to the covariance $\mathbf{\Sigma}_c$ [17] as follows:

$$\widetilde{\mathbf{F}}_c = \mathbf{\Sigma}_c^{-1/2} \left[\sum\nolimits_t \gamma_c(t)\,(o_t - \mu_c)\right] \tag{7}$$

Note that acoustic observation $\mathbf{x}_t$ for the DNN and o_t for the total variability model are typically different. The former generally span a wider context that is not required in the latter.

3. CONTENT-AWARE LOCAL VECTOR

Since senones are endowed with clear phonetic interpretations, in this paper, we propose to extract content-aware local vectors using the clustered senonic posterior probabilities for frame alignment.

3.1. Senone clustering

Different from the hand-crafted grouping method presented in [9], [10] and [11], in this paper, the Gaussian-like units are clustered agglomeratively based on the minimum log-likelihood cost [18]. Given a background training dataset of R speech utterances, the likelihood function is defined as

$$l = \prod_{r}^{R} \prod_{c}^{C} \prod_{t}^{T_r} \mathcal{N}(o_{r,t}|\mu_c, \mathbf{\Sigma}_c)^{\gamma_c(r,t)} \tag{8}$$

Taking the logarithm of (8), the log-likelihood is given by

$$\begin{aligned} L = &\sum_{c=1}^{C} N_c log \frac{1}{(2\pi)^{D/2}\,|\mathbf{\Sigma}|^{1/2}} - \frac{1}{2}\sum_{c=1}^{C} tr\left(\mathbf{\Sigma}_c^{-1}\mathbf{S}_c\right) \\ &+ \sum_{c=1}^{C}\left(\mu_c^{\mathsf{T}}\mathbf{\Sigma}_c^{-1}\mathbf{F}_c\right) - \frac{1}{2}\sum_{c=1}^{C}\left(\mu_c^{\mathsf{T}}\mathbf{\Sigma}_c^{-1}\mu_c N_c\right) \end{aligned} \tag{9}$$

where N_c, $\mathbf{F}_c$ and $\mathbf{S}_c$ are the zero, first and second order statistics accumulated across R utterances as follows:

$$N_c = \sum_{r=1}^{R} \sum_{t}^{T_r} \gamma_c(r,t) \tag{10}$$

$$\mathbf{F}_c = \sum_{r=1}^{R} \sum_{t=1}^{T_r} \gamma_c(r,t)\, o_{r,t} \tag{11}$$

$$\mathbf{S}_c = \sum_{r=1}^{R} \sum_{t=1}^{T_r} \gamma_c(r,t)\, o_{r,t} o_{r,t}^{\mathsf{T}} \tag{12}$$

where $o_{r,t}$ is the t-th frame from the r-th utterance; $\gamma_c(r,t)$ is its posterior occupancy on the c-th senone; and T_r denotes the number of frames in the r-th utterance. Note that the first-order statistics $\mathbf{F}_c$ in (11) is calculated without centering or whitening as what was done for $\widetilde{\mathbf{F}}_c$ in (7). During the clustering, every unit is merged with the rest and the log-likelihoods are calculated given the merge. The one results in maximum log-likelihood (i.e., minimum log-likelihood cost) is selected. Let ς be the parameters (mean and covariance) of the Gaussian-like unit. The parameters are updated as:

$$\varsigma_{cc'} = \frac{N_c \varsigma_c + N_{c'} \varsigma_{c'}}{N_c + N_{c'}} \tag{13}$$

where two Gaussian-like units c and c' are merged to form the merged unit cc'. Similarly, the statistics of the merged unit is obtained by pooling those from the component units

$$\kappa_{cc'} = \kappa_c + \kappa_{c'} \tag{14}$$

where κ represents the set of statistics as $\kappa = \{N, \mathbf{F}, \mathbf{S}\}$. The merging is done iteratively until the target number of clusters is obtained. With clustering, the senones of close Markov states are further tied and each cluster can be regarded as the representation of a certain phonetic class. Notice that our implementation is different from that of [18] where the likelihood were calculated on the frame instead of the statistics level. Furthermore, we resort to local variability modeling in the i-vector extraction which allows content matching in the PLDA scoring.

3.2. Content-aware local vector estimation

Assuming K clusters are trained, by treating the senones as Gaussian-like units, K local variability vectors can be estimated. Denoting the set of senones in the k-th $(k = 1, ..., K)$ cluster as $\mathcal{S}_k = \left\{ s_{i_1^k}, ..., s_{i_j^k}, ..., s_{i_{C_k}^k} \right\}$ where C_k denotes the number of senones in the k-th cluster and i_j for $j = 1, ..., C_k$ denotes the index of the j-th senone, the local variability model can be described as

$$\mathbf{m}_{r,k} = \mu_k + \mathbf{V}_k \mathbf{w}_{r,k} \tag{15}$$

where μ_k is the mean vector of the k-th cluster; $\mathbf{V}_k$ is the corresponding local variability loading matrix, and $\mathbf{w}_{r,k}$ is its local latent variable. The posterior mean of $\mathbf{w}_{r,k}$ is defined as the content-aware local vector, computed as:

$$\phi_k = \mathbf{L}_k^{-1} \sum_{j=1}^{C_k} \left[\mathbf{V}_{k,i_j} \right]^{\mathsf{T}} \widetilde{\mathbf{F}}_{k,i_j} \tag{16}$$

where $\mathbf{L}_k$ is the posterior precision matrix of $\mathbf{w}_k$ calculated as

$$\mathbf{L}_k = \mathbf{I} + \sum_{j=1}^{C_k} N_{k,i_j} \left[\mathbf{V}_{k,i_j} \right]^{\mathsf{T}} \mathbf{V}_{k,i_j} \tag{17}$$

For simplicity, the utterance index r is dropped from (16) and (17). Since the senone clusters are phonetic related, the local vectors therefore representing the session variability under the respective phonetic context, named as the content-aware local vectors.

4. CONTENT-AWARE CHANNEL COMPENSATION

For the r-th speech utterance of speaker s, a local composite vector can be obtained by concatenating its content-aware local vectors as $\phi_{s,r} = \left[\phi_{1,s,r}^{\mathsf{T}}, ..., \phi_{K,s,r}^{\mathsf{T}} \right]^{\mathsf{T}}$. A PLDA model is trained on these vectors to get rid of the influence of channel variability on speaker comparison [5]:

$$p(\phi_{s,r}) = \mathcal{N}\left(\omega, \mathbf{\Phi}\mathbf{\Phi}^{\mathsf{T}} + \mathbf{\Omega}\mathbf{\Omega}^{\mathsf{T}} + \mathbf{\Lambda} \right) \tag{18}$$

where ω, $\mathbf{\Phi}$, $\mathbf{\Omega}$ and $\mathbf{\Lambda}$ are the global mean vector, speaker subspace, channel subspace and residual covariance. The parameters are trained using similar EM algorithm presented in [5]. In terms of the local vectors, the PLDA model can be decomposed as

$$\phi_{k,s,r} = \omega_k + \mathbf{\Phi}_k \mathbf{h}_s + \mathbf{\Omega}_k \mathbf{x}_{k,s,r} + \epsilon_k \tag{19}$$

where ϵ denotes the residual and the subscript k indicates the submatrices and subvectors corresponding to the k-th local vector. From this perspective, the speaker information in the local vectors can be compared under the same phonetic context. On the other hand, with the speaker and channel subspaces being trained on the composite vector, the correlations between phonetic classes are also considered in the form of covariance as:

$$\widetilde{\mathbf{\Lambda}} \leftarrow \mathbf{\Phi}\mathbf{\Phi}^{\mathsf{T}} + \mathbf{\Omega}\mathbf{\Omega}^{\mathsf{T}} + \mathbf{\Lambda} \tag{20}$$

5. EXPERIMENTS

5.1. Experiment setup

In our experiments, the DNN was trained on Fisher and Switchboard dataset using the Kaldi toolkit [19]. The feature vectors for the DNN were 40-dimensional filter bank coefficients with first and second derivatives appended, leading the dimension of the features to be 120. The input feature is concatenated with 5 frames before and after it as the input to the network. The structure of the network is 1320-2048×5-556 with 556 to be the number of senones. After eliminating 20 senones associated with silence, laughter, noise and OOV, 536 valid senones are used for variability modeling.

The acoustic features for speaker modeling was 19-dimensional MFCC feature with its first and second derivatives appended on which RASTA and CMVN were applied. Given the DNN posteriors, the parameters of the UBM with full covariance was calculated with data selected from NIST SRE'04, 05, 06 and Switchboard. The total and local variability models were trained with the data drawn from NIST SRE'04, 05, 06 and Switchboard. Both the UBM and variability models were trained in a gender-dependent manner. The dimension of the DNN i-vector was set to 400. The PLDA consisted of a speaker subspace of rank 200 and a full residual covariance. For local variability model, the senones were clustered to 13 groups and each group was associated with a local vector of dimension 50, resulting in a $13 \times 50 = 650$-dimensional content-aware (CA) local vector for each speech utterance. The PLDA on the CA local vector was composed of a speaker subspace of rank 200, a channel subspace of rank 650 and a diagonal residual covariance. The performance was evaluated based on the equal-error-rate (EER) and the minimum detection cost function (DCF) [20] defined for SRE'08.

Table 1. Performance (EER(%)/mDCF08×100) of DNN i-vector (DNN i-vec) and content-aware local vectors (CA-lv) on condition 6 of *short2-short3*, *short2-10sec*, *10sec-10sec* of NIST SRE'08 trails.

		male	female
shor2-short3	# non / # tar	11,637 / 874	21,563 / 1,802
	DNN i-vec	**4.25** / **22.44**	**6.10** / **32.76**
	CA-lv	4.63 / 25.23	6.62 / 35.84
shor2-10sec	# non / # tar	6,766 / 508	12,159 / 1,001
	DNN i-vec	9.28 / 46.28	12.63 /61.91
	CA-lv	**8.49** / **44.92**	**11.78** / **58.21**
10sec-10sec	# non / # tar	6,528 / 493	11,899 / 979
	DNN i-vec	19.95 / 84.81	22.67 /89.17
	CA-lv	**14.04** / **72.09**	**16.84** / **72.70**

Table 2. Performance (EER(%)/mDCF08×100) of DNN i-vector (DNN i-vec) and content-aware local vectors (CA-lv) on RSR2015 part I : *text-constrained* and *text-dependent* trials respectively.

		male	female
text-constrained	# non / # tar	215,656 / 4,404	189,940 / 4,135
	DNN i-vec	9.38 / 4.83	13.22 / 6.14
	CA-lv	**7.69**/**4.10**	**11.49** /**5.89**
text-dependent	# non / # tar	387,230 / 8,419	437,631 / 8,931
	DNN i-vec	5.31 / 2.42	6.03 /**2.99**
	CA-lv	**4.68** / **2.22**	**5.65** / 3.02

5.2. Results

We experimented on the *text-independent*, *text-constrained* and *text-dependent* tasks in the following.

Text-independent experiments: The experiments were carried out on NIST SRE'08 where two nominal durations were included, i.e., 10 seconds and 2.5 minutes after voice activity detection (VAD). The PLDA models on the DNN i-vector and the CA local vector were trained with the telephone data from NIST SRE'04, 05 and 06 in a gender-dependent manner. Three tasks were tested: *short2-short3*, *short2-10sec* and *10sec-10sec*. Since the *short2* and *short3* utterances of 2.5 minutes are supposed to contain almost all the lexical variability while the 10-second utterances only contain a part, content mismatch is a problem for the *short2-10sec* and *10sec-10sec* tasks but not the *short2-short3* task. The performances of DNN i-vector and CA local vectors in the three tasks are given in Table 1. We can see that the CA local vectors outperform the DNN i-vector in handling the content mismatch in the *short2-10sec* and *10sec-10sec* tasks by comparing the speakers in the utterances based on the phonetic classes. No performance gain was achieved on the *short2-short3* task where there is no content mismatch problem.

Text-constrained experiments: The RSR2015 dataset [21] was used in the experiments, which consists of 300 speakers and is divided into background (*bkg*), development (*dev*) and evaluation (*eval*) subsets. Part I of the *dev* subset was used in our experiments whose lexical content is constrained to 30 short sentences drawn from TIMIT dataset with an average nominal duration of 1.25s after VAD. Text constraint is reflected in that the text of every utterance is one of the 30 sentences. The testing trials were extracted from the configuration "3sessall_dev" by deleting the 16-30 utterances from enrollment and 1-15 from test. This makes the text in the test utterance unseen by the enrollment utterances. The speeches were downsampled to 8 kHZ. The PLDA model was trained in a gender-independent manner, using all the utterances in part I of the speakers in the *bkg* subset.

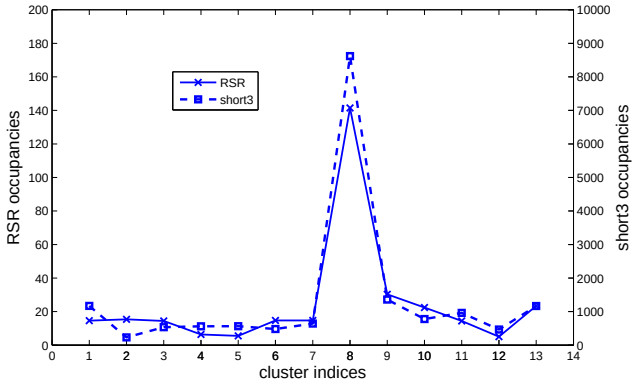

Fig. 3. Occupancies in the 13 clusters of one utterance from *short3* and RSR Part I respectively.

Text-dependent experiments: We adopted the configuration "3sess-pwd_dev" of part I. Each trial is based on a certain sentence, whose recordings of sessions 1, 4 and 7 are used for speaker enrollment while those of the rest sessions for test. The PLDA models in the *text-constrained* experiment were used.

Table 2 presents the performance comparisons of DNN i-vector and CA local vectors on the *text-constrained* and *text-dependent* tasks where the CA local vectors perform superiorly to the DNN i-vector. Since the lexical contents for test are unseen by the enrollment utterances in the *text-constrained* task, the superiority of the CA local vectors owes to its speaker comparison with respect to the phonetic classes, showing the capability of them in coping with the content mismatch.

In the *text-dependent* task, the sentences in the test and enrollment utterances are the same. Like the *short2-short3* task of NIST SRE'08, there is no content mismatch. However, it's interesting to see that the CA local vectors outperform the DNN i-vector. This further validates the effectiveness of the CA local vectors for short duration utterances. Fig. 3 gives the occupancies of two randomly chosen utterances from *short3* and RSR Part I on the 13 clusters respectively. We can see that the occupancies of the utterance from RSR on all of the 13 clusters are small (about 1/50 of those from the *short3* utterance). As a result, the variability vector estimated for the RSR utterance is more sensitive to the occupancies. The local vector estimated in a cluster catches the minor variability in the cluster which cannot be modeled by an i-vector. Also shown in Fig. 3, given an utterance from RSR or *short3*, none of the occupancies in the clusters is zero, making it unnecessary to select the local vectors for content matching as what we did in [13].

6. CONCLUSION

We proposed to estimate the content-aware local vectors on classes of senones which are modeled with a deep neural network. The local vectors provide the representations of the session variability contained in the phonetic contents. PLDA was applied as the channel compensation technique which can compare the local vectors in a content-aware manner while considering their correlations. Our experiments show that the content-aware local vectors outperform the DNN i-vector in the trials where short utterances are included. This results show the potential of the proposed local vector in coping with the content mismatch problem in short utterances.

7. REFERENCES

[1] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 13, pp. 19 – 41, 2000.

[2] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12 – 40, 2010.

[3] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[4] P. Kenny, M. Mihoubi, and P. Dumouchel, "New map estimators for speaker recognition," in *Proc. INTERSPEECH*, 2003.

[5] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. ICCV*, 2007, pp. 1–8.

[6] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep neural networks for extracting baum-welch statistics for speaker recognition," in *Proc. Odyssey*, 2014, pp. 293–298.

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

[8] T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. K., and P. Dumouchel, "Text-dependent speaker recognition using PLDA with uncertainty propagation," in *Proc. INTERSPEECH*, 2013, pp. 3684–3688.

[9] L. Chen, K. A. Lee, B. M., W. Guo, H. Li, and L. R. Dai, "Local variability modeling for text-independent speaker verification," in *Proc. Odyssey: Speaker and Language Recognition Workshop*, 2014, pp. 54–59.

[10] L. Chen, K. A. Lee, B. Ma, W. Guo, H. Li, and L. R. Dai, "Exploration of local variability in text-independent speaker verification," *Journal of Signal Processing Systems*, pp. 1–12, 2015.

[11] L. Chen, K. A. Lee, L. R. Dai, and H. Li, "Quasi-factorial prior for i-vector extraction," *accepted by Signal Processing Letters, IEEE*, 2015.

[12] P. Rose, *Forensic Speaker Identification*, Taylor & Francis, London, 2012.

[13] L. Chen, K. A. Lee, B. Ma, W. Guo, H. Li, and L. R. Dai, "Phone-centric local variability vector for text-constrained speaker verification," in *Proc. INTERSPEECH*, 2015, pp. 229–233.

[14] M. Y. Hwang, X. Huang, F. Alleva, et al., "Predicting unseen triphones with senones," *Speech and Audio Processing, IEEE Transactions on*, vol. 4, no. 6, pp. 412–419, 1996.

[15] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[16] A. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 14–22, 2012.

[17] P. Kenny, "A small footprint i-vector extractor," in *Proc. Odyssey: Speaker and Language Recognition Workshop*, 2012.

[18] S. Cumani, P. Oldrick, and F. Kulsoom, "Speaker recognition by means of acoustic and phonetically informed gmms," in *Proc. INTERSPEECH*, 2015, pp. 200–204.

[19] D. Povey and et al., "The kaldi speech recognition toolkit," in *Proc. ASRU*, 2011, pp. 200–204.

[20] N. Brümmer and J. Preez, "Application-independent evaluation of speaker detection," *Computer & Speech Language*, vol. 20, no. 2, pp. 230–275, 2006.

[21] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Text-dependent speaker verification: Classifiers, databases and RSR2015," *Speech Communication*, vol. 60, pp. 56–77, 2014.