Multi-Task Learning for Text-dependent Speaker Verification

Nanxin Chen Yanmin Qian Kai Yu

Key Lab. of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering
SpeechLab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China

bobchennan@gmail.com, yanminqian@sjtu.edu.cn, kai.yu@sjtu.edu.cn

Abstract

Text-dependent speaker verification uses short utterances and verifies both speaker identity and text contents. Due to this nature, traditional state-of-the-art speaker verification approaches, such as i-vector, may not work well. Recently, there has been interest of applying deep learning to speaker verification, however in previous works, standalone deep learning systems have not achieved state-of-the-art performance and they have to be used in system combination or as tandem features to obtain gains. In this paper, a novel multi-task deep learning framework is proposed for text-dependent speaker verification. First, multi-task deep learning is employed to learn both speaker identity and text information. With the learned network, utterance level average of the outputs of the last hidden layer, referred to as *j-vector*, means joint-vector, is extracted. Discriminant function, with classes defined as multi-task labels on both speaker and text, is then applied to the j-vectors as the decision function for the *closed-set recognition*, and Probabilistic Linear Discriminant Analysis (PLDA), with classes defined as on the multi-task labels, is applied to the j-vectors for the *verification*. Experiments on the RSR2015 corpus showed that the j-vector approach leads to good result on the evaluation data. The proposed multi-task deep learning system achieved 0.54% EER, 0.14% EER for the closed-set condition.

Index Terms: deep neural network, multi-task learning, speaker verification, discriminant analysis, probabilistic linear discriminant analysis, deep learning

1. Introduction

Recently there has been wide interests in text-dependent speaker verification for applications such as automated password reset, audio command systems, speech fingerprint, and user identification. The task of speaker verification is to identify the person who is speaking according to some characteristics of their voices. When the process includes the option of declaring that the test utterance does not belong to any of the known (registered) speakers, then it is referred to as open-set speaker recognition. Also according to whether the text of test speech is the same as that of enrollment speech, two kinds of speaker verification systems are involved: text-dependent and text-independent. The main advantage of text-dependent speaker verification systems is that it restricts the content of speech text, so the knowledge of lexicon can be learned in models, which leads to higher accuracy in real application. Besides the speech duration for registration is crucial in realistic scenario, which has a great impact on user experience. Accordingly the text-dependent systems, which only need short utterances, are more of practical interests. However, short duration also makes it a challenging problem. Traditional state-of-the-art speaker verification approaches, such as GMM-UBM or i-vector, may not work well in this case[1, 2, 3]. Therefore, new algorithms are needed for text-dependent speaker verification.

Deep learning recently attracts more and more interests in machine learning community. It has also achieved great success in speech processing, such as speech recognition [4], synthesis [5, 6] and enhancement [7, 8]. There are also some works proposed which was focusing on how to use it for speaker verification. Previous works of neural networks using for speaker verification can be traced back to the last century, including combining the property of decision trees and feed-forward neural networks [9, 10, 11, 12], auto-associative neural network (AANN) model [13, 14] and non-linear discriminant analysis (NLDA) [15], and so on. These works mostly employed shallow architecture of neural network. In recent years, people started using *deep neural networks* (DNN) for speaker verification applications. Our previous work proposed to use tandem deep features, which are extracted from deep neural networks, for building GMM-UBM text-dependent speaker verification systems[16]. d-vectors[17], which are extracted from a deep neural network, could get good performance, and get additive improvement after combined with i-vectors[18]. Other works such as Nicolas Scheffer's work [19] attempts to use phone-discriminant DNN to extract features to solve content mismatch issue. Yun Lei's work [20] use ASR-DNN system to produce frame posteriors, which improves the performance of the i-vector system. However in all of these works, deep learning still has to be used in system combination or as tandem features to get better and more satisfactory results than the traditional methods.

In this work multi-task learning is used which could learn the knowledge from both the speakers and text phrases simultaneously in just one framework, to improve the capacity of deep neural networks. A new discriminative feature, defined as j-vector, is extracted from these multi-task DNNs. *Joint probabilistic linear discriminant analysis* (PLDA), with classes defined as multi-task labels on both speaker and text, is then applied to the j-vectors as the decision function. Experiments show that compared to the previous standalone systems, the proposed multi-task deep learning framework can achieve better results on the RSR2015 corpus[1].

The remainder of this paper is organized as follows. Section 2 reviews two recent works on text-dependent speaker verification which utilized the deep neural networks: tandem deep features and d-vectors. Section 3 describes the novel multi-task

This work was supported by the Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning, the China NSFC project No. 61222208 and JiangSu NSF project No. 201302060012.

learning method in details. Section 4 gives the experimental results, comparison and detailed analysis. The whole work is summarized in Section 5.

2. Deep Learning for Text-Dependent Speaker Verification

In this section two recent works on text-dependent speaker verification using deep neural networks are reviewed: tandem deep features and d-vectors.

2.1. Tandem deep features

Tandem deep features, which is proposed in our previous work [16], utilized the deep neural network as a feature extractor for speaker verification. In this approach raw spectral features are spanned in a context window (e.g. 11 frames, 5 frames on each side) and fed into a refined deep neural network to generate new features. Once the network is trained, the outputs from the hidden layers are utilized as high-level features, optionally combined with the original spectral features, to build GMM-UBM speaker verification systems. Three different deep features are proposed in that work, including RBM, speaker-discriminant or phone-discriminant neural network based features. All of them get significant improvements individually when compare to the spectral feature based systems. Moreover after feature combination, better performance can be obtained.

2.2. d-vectors

Motivated by the powerful acoustic modelling capability and great success of deep neural networks in speech recognition[4], some researchers proposed to use DNN for speaker identification directly. Different from the above described method which uses a DNN to extract deep features [21, 16], here people extract the speaker identity representations directly from a neural network, which is similar as the i-vector idea. In Google's previous work, a DNN is trained to map frame-level features in a given context to the corresponding speaker identity target. During enrollment, the speaker model is computed as the average of outputs derived from the last DNN hidden layer, which are defined as a deep vector or "d-vector"[17]. In the evaluation phase, decisions are made according to the (cosine) distance between the target d-vector and the test d-vector, which is similar as the normal i-vector speaker verification systems.

3. Multi-Task Deep Learning for Text-Dependent Speaker Verification

A new multi-task deep learning model is introduced in this section, including training, enrollment and test process.

3.1. Multi-Task Learning of Deep Neural Network

The intuition behind this multi-task training process is that directly recognizing speaker seems to be hard but in reality different speakers have their own style on each syllable or word. Thus the optimization towards text targets, for instance, phone-discriminant or phrase-discriminant, gives some hints to recognize speakers as well. When DNN is used, with the huge number of parameters in neural network, the DNN model can be learned together to discriminate both texts and speakers. This is the same idea as multi-task learning.

As shown in Fig 1, for simplicity the sum of the two original loss function $C_1(\boldsymbol{y}_1, \boldsymbol{y}_1{}')$, $C_2(\boldsymbol{y}_2, \boldsymbol{y}_2{}')$ is used as the total loss

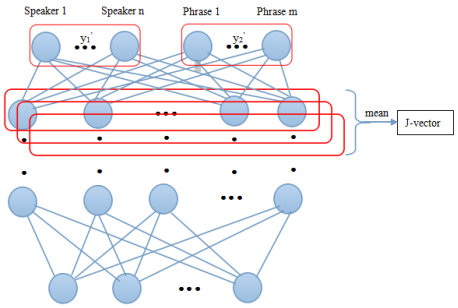

Figure 1: Multi-task Deep Learning for Text Dependent Speaker Verification

function:

$$C([\boldsymbol{y}_1, \boldsymbol{y}_2], [\boldsymbol{y}_1{}', \boldsymbol{y}_2{}']) = C_1(\boldsymbol{y}_1, \boldsymbol{y}_1{}') + C_2(\boldsymbol{y}_2, \boldsymbol{y}_2{}') \quad (1)$$

where C_1, C_2 are two cross-entropy criteria for speakers and texts. $\boldsymbol{y}_1$, $\boldsymbol{y}_2$ indicate the true labels for speakers and texts individually, while $\boldsymbol{y}_1{}'$, $\boldsymbol{y}_2{}'$ are the outputs of the two targets respectively. According to the linearity of the gradient, the gradient of each parameter can be calculated respectively, and new parameters on common layers can be updated by the gradient from the sum of two loss functions. The learning rate is reduced when the classify accuracy of the two tasks does not get improvement either. Multi-task learning avoids over-fitting for DNN training, and enhances the functionality of DNN nodes.

Once the multi-task neural network training process is done, both of the output layers are removed, and the rest of the neural network (common hidden layers) is used to extract the speaker-text joint representation (just using the outputs of the last hidden layer). The average of these outputs in the same audio is defined as *j-vectors* (joint vector) for that audio in this paper. Then the new proposed j-vectors could be utilized similarly as the d-vectors[17].

3.2. Register and Verification

j-vectors, as more discriminative and compact representation, can use several different back-end classifications, such as cosine similarity, linear discriminant analysis (LDA) [22, 23], and probabilistic linear discriminant analysis (PLDA) [24, 25]. Previous work on d-vector [17] only used cosine similarity for verification, however other classifiers are investigated here in detail. The following experiments show that joint discriminant function and PLDA give much better results than cosine similarity metric which was utilized in d-vector approach.

3.2.1. Joint Gaussian Discriminant Function

Linear discriminant analysis (LDA) [22, 23] provides good generalization capability even with limited number of training samples. One of the advantages of this model is that LDA attempts to define new special axes that minimize the intra-class variance caused by channel effects, and to maximize the variance between classes. Due to these reasons it was used on many tasks relate to speaker recognition [23, 26]. It assumes that each class density can be modelled as a multivariate gaussian:

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{\frac{p}{2}}|\boldsymbol{\Sigma}_k|^{\frac{1}{2}}} \mathrm{e}^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k{}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)} \quad (2)$$

where $\mathbf{\Sigma}_k$ and $\boldsymbol{\mu}_k$ is the covariance and mean for class k, p is the dimension of the vector. LDA model assumes $\mathbf{\Sigma}_k = \mathbf{\Sigma}, \forall k$.

Our experiments show that the outputs of the multi-task learning network has strong discriminant property. The transformation matrix $\boldsymbol{w}$ is estimated by the development data. The estimation of transformation matrix $\boldsymbol{w}$ in LDA does not leads to improvement. So without the consideration of $\boldsymbol{w}$, according to the linear discriminant function analysis, the discriminant function for class k could be wrote as

$$\mathrm{df}_k(\mathbf{x}) = -\frac{1}{2} \times (\mathbf{x} - \boldsymbol{\mu}_k)^\top \mathbf{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}_k) \tag{3}$$

For the closed-set case, the equation can be expanded and the first term $\mathbf{x}^\top \mathbf{\Sigma} \mathbf{x}$ gets the same value of all classes. The rest terms could write as the linear expression (GDF):

$$\mathrm{df}_k^{'}(\mathbf{x}) = (\mathbf{\Sigma}^{-1} \boldsymbol{\mu}_k)\mathbf{x} + (-\frac{1}{2} \boldsymbol{\mu}_k{}^\top \mathbf{\Sigma}^{-1} \boldsymbol{\mu}_k) \tag{4}$$

And the probability could be estimated by Bayes theorem. Here an important characteristics is to define the class as the multi-task class considering both speaker and text phrase information, similar to multi-task learning of deep neural networks.

3.2.2. Probabilistic Linear Discriminant Analysis

LDA uses gaussian mixture model, which can be regarded as a latent variable model where the observed node $\mathbf{x}$ represents the example and the latent variable $\boldsymbol{\mu}_k$ is the center of a mixture component representing class k. The class-conditional distribution is $P(\mathbf{x}|k) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\phi})$ where $\boldsymbol{\phi}$ is shared by all classes. PLDA[24, 25] propose to make the latent variable prior continuous. Particularly, to enable efficient inference and closed-form training, a Gaussian prior is imposed: $P(\boldsymbol{\mu}_k) = \mathcal{N}(\boldsymbol{\mu}_k|\boldsymbol{m}, \boldsymbol{\phi}_b)$

It is worth noted that, LDA or PLDA only with single speaker labels always obtain degraded performance in our experiments, and hence they are not discussed further.

4. Experiments

In this section experiments and comparison details are given for the baselines and proposed new model.

4.1. Experimental Setup and Baseline System

To evaluate and compare different systems, RSR2015 database[1] which was released by I2R, is used to evaluate the performance. The corpus contains audios recording from 300 people, which includes 143 female and 157 male speakers that are between 17 to 42 years old. The whole set is divided into background (bkg), development (dev) and evaluation (eval) subsets. All audios are recorded using three portable devices, into nine sessions. Each session contains thirty short phrases. The average duration of these audio is 3.2 seconds.

When testing, a speaker is enrolled with 3 utterances of the same phrase. The corresponding test utterances are also of the same phrase, however all utterances in a trial come from different sessions and are taken from the eval set. The task concerns on both the phrase content and speaker identification. Three different errors are taken into consideration: non-target speaker, content mismatch, and both.

The baseline system is constructed using the gender-independent GMM-UBM approach. 39-dimensional PLP features with short-term mean and variance normalization is used as the spectral features in our experiments. An energy-based Voice Activity Detection (VAD) is utilized to detect the speech segments, and a gender-independent UBM of 512-components is trained using both bkg and dev data. In test data set there are 19052 tests for true speaker and 1548956 tests for imposture. Slightly different from our previous work in [16], which was also tested on the RSR2015 text-dependent speaker verification task, the score normalization post-processing, ZNORM [27] is applied. The EER and minDCF of the baseline system are shown in the first line of Table 1. It can be seen that the EER is relatively low compared to the text-independent tasks, which is also relative hard to improve.

Table 1: Performance of previous deep learning approaches

Feature	Classifier	EER(%)	minDCF
PLP	GMM-UBM	0.86	0.048
Tandem		0.69	0.037
d-vector	Cosine Sim.	21.05	0.818

To do better comparison with previous work using deep learning, we also construct the systems using Tandem deep features which was proposed in our previous work [16] [1], and the d-vectors approach which was proposed in Google's work [17].

Speaker based DNN classifiers are needed to be built in all these previous work [16, 17]. The deep models are all built with 4 hidden layers with 1024 nodes per layer, and a context window of 31 frames (15 frames on each side) 39-dim PLP is used as the neural network inputs. The bkg and dev data are used in the network training, and 194 classes (194 speakers in bkg and dev set) are used in the speaker DNN construction. The RBM pre-training is used for model initialization and SGD based back-propagation is applied in DNN training. The learning rate annealing and early stopping strategies are used in the BP processing and the DNN is fine-tuned with cross-entropy objective function , along with a L2-norm weight-decay term of coefficient 10^{-6}.

4.1.1. Tandem deep features

Speaker-discriminant DNN is used in this work which show the best performance among individual deep features in our previous works [16]. For each speech frame, PCA is applied on the outputs of hidden layers and reduce the dimension to 39 dims. This deep features can be connected with the original PLP feature to form the new concatenated Tandem deep features. After the tandem deep feature extraction, the normal GMM-UBM framework is implemented. The ZNORM approach is utilized as the normal baseline described above [2].

The performance of the Tandem deep features is illustrated as the second line of Table 1.

4.1.2. d-vectors

This speaker-discriminant DNN is used to extract the d-vector as Google's previous work in [17]. The accumulated outputs of the last hidden layer are taken as a new speaker representation. The outputs of the last hidden layer using standard feed-forward propagation in the trained DNN, are accumulated with L2-normalization to form d-vector. The final d-vector representation of the speaker s is derived by averaging all d-vectors corresponding from the utterances of the same speaker s. In test

[1] Based on conclusion in our previous work [16] which shows speaker-discriminant feature gets the best position, we just built the system using speaker-discriminant tandem deep features.

[2] Less hidden layers and less gaussian mixtures are used here compared to our work in [16]

period d-vector is extracted from the test utterance, and the verification decision is made according to the cosine similarity as the work in [17].

The performance of the d-vectors is illustrated as the last line of Table 1. In this RSR2015 task, very bad performance is observed by using the d-vector with cosine similarity directly.

4.2. Evaluation of j-vectors

For our newly proposed multi-task learning of deep neural network, DNN are built with 4 hidden layers and 1024 nodes in each layer. Most of the training configurations are the same as the previous descriptions in baseline. After the multi-task DNN training, the j-vectors are extracted from the outputs of the last hidden layer as described in Section 3.

The PLDA model is then trained using the j-vectors. The class defined in the PLDA is the multi-task label on both the speaker and text. For each test audio the j-vectors are extracted using the same steps and then the log likelihood from PLDA algorithm is used to distinguish among different models. The with-in class covariance smoothing parameter [3] is set to 0.75 and then the PLDA model is estimated with 20 iterations. For easy comparison, the normal cosine similarity based decision function is also used. Besides the proposed multi-task learned deep neural network, other neural networks are also investigated here, including deep Restricted Boltzmann machine (denoted as r-vector), and speaker-discriminant deep neural network (just the d-vector system). Similarly the discriminant function utilized systems are also constructed.

The performance comparison is presented in Table 2. Here GDF indicates gaussian discriminant function, as mentioned in section 3.2.1. Obviously j-vectors are superior than the r-vector or d-vector extracted from other neural networks, no matter which classifier is used in the system construction. A significant improvement is obtained when using the GDF or PLDA instead of the cosine similarity in all types of vectors, especially within our proposed novel j-vectors framework. The Detection error trade-off (DET) curve is illustrated in figure 2 while only non-target speakers are considered as impostor.

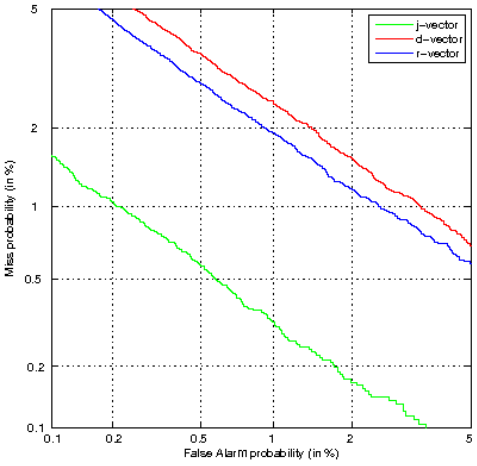

Figure 2: DET curves for RSR2015 Part I evaluation set. In all trials, target and impostor speaker pronoun the correct text

[3] In order to get a good estimate of the within-class covariance, the production of this parameter and the between-class covariance is adding to the within-class covariance

Table 2: Performance for different deep learning systems

Feature	Classifier	EER	minDCF
r-vector	Cosine Sim.	17.43	0.684
	Joint GDF	0.80	0.037
	Joint PLDA	1.47	0.065
d-vector	Cosine Sim.	21.05	0.818
	Joint GDF	0.71	0.033
	Joint PLDA	1.62	0.070
j-vector	Cosine Sim.	9.85	0.466
	Joint GDF	**0.14**	**0.007**
	Joint PLDA	0.54	0.027

To further demonstrate the superiority of the j-vectors, we also do some testing with the utterances from the unseen speakers, which means that there are some impostor speakers non-existing during the enrollment. To imitate this situation, a portion (e.g. $\frac{1}{3}$ and $\frac{1}{5}$) of the enroll speakers is removed. Then we still test whether test segments of these removed speakers matches other enroll speakers. So the speakers in the removing part are unseen in the enrollment, but tested on the test process (using other models). The same evaluation process is implemented as before.

The results in this condition are illustrated in Table 3. In all the cases, the proposed j-vector obtained the best performance. Although there is a slightly performance decline when compared to the results in Table 2 when using gaussian discriminant function (GDF), the j-vector with these model is still much better than the baseline systems. These experiments also demonstrate the generalization of the proposed j-vectors.

Table 3: Performance under unseen speakers conditions

Unseen Speakers Ratio		1/5		1/3	
Feature	Classifier	EER	minDCF	EER	minDCF
r-vector	Cosine Sim.	20.68	0.820	20.71	0.818
	Joint GDF	1.33	0.062	1.63	0.076
	Joint PLDA	1.42	0.066	1.65	0.073
d-vector	Cosine Sim.	15.75	0.644	15.75	0.654
	Joint GDF	1.43	0.063	1.78	0.079
	Joint PLDA	1.56	0.063	1.65	0.067
j-vector	Cosine Sim.	9.65	0.463	9.64	0.464
	Joint GDF	**0.47**	0.033	0.58	0.050
	Joint PLDA	0.50	**0.022**	**0.50**	**0.024**

5. Conclusion

This paper proposed a novel framework using deep learning technology for text-dependent speaker verification. A multi-task deep learning framework is described for extracting useful knowledge from multi-level. First, multi-task deep learning is employed to learn both speaker identity and text information. With the learned network, utterance level average of the outputs of the last hidden layer, referred as *j-vector*, are extracted. Finally gaussian discriminant function (GDF) or probabilistic linear discriminant analysis (PLDA) is applied to the j-vectors as the decision function with the joint classes defined as multi-task labels on both the speaker and text. Experiments show that our proposed j-vector approach gets a large improvement compared to the other recently proposed deep learning approaches, such as d-vectors and tandem deep features, in the text-dependent speaker verification. The proposed system achieves 0.54% EER in RSR2015 corpus, 0.14% EER for closed-set evaluation.

6. References

[1] A. Larcher, K.-A. Lee, B. Ma, and H. Li, "Rsr2015: Database for text-dependent speaker verification using multiple pass-phrases." in *Proc. InterSpeech*, 2012.

[2] H. Aronowitz, "Text dependent speaker verification using a small development set," in *Odyssey 2012-The Speaker and Language Recognition Workshop*, 2012.

[3] A. Larcher, P. Bousquet, K. A. Lee, D. Matrouf, H. Li, and J.-F. Bonastre, "I-vectors in the context of phonetically-constrained short utterances for speaker verification," in *Proc. ICASSP*. IEEE, 2012, pp. 4773–4776.

[4] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and K. Brian, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[5] H. Ze, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. ICASSP*. IEEE, 2013, pp. 7962–7966.

[6] Y. Fan, Y. Qian, F. Xie, and F. K. Soong, "Tts synthesis with bidirectional lstm based recurrent neural networks," in *Proc. InterSpeech*, 2014.

[7] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder." in *Proc. InterSpeech*, 2013, pp. 436–440.

[8] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *Signal Processing Letters, IEEE*, vol. 21, no. 1, pp. 65–68, 2014.

[9] A. Sankar and R. J. Mammone, "Speaker independent vowel recognition using neural tree networks," in *Neural Networks, 1991., IJCNN-91-Seattle International Joint Conference on*, vol. 2. IEEE, 1991, pp. 809–814.

[10] M. G. Rahim, "A neural tree network for phoneme classification with experiments on the timit database," in *Proc. ICASSP*, vol. 2. IEEE, 1992, pp. 345–348.

[11] H.-S. Liou and R. J. Mammone, "A subword neural tree network approach to text-dependent speaker verification," in *Proc. ICASSP*, vol. 1. IEEE, 1995, pp. 357–360.

[12] K. Farrell, S. Kosonocky, and R. Mammone, "Neural tree network/vector quantization probability estimators for speaker recognition," in *Neural Networks for Signal Processing [1994] IV. Proceedings of the 1994 IEEE Workshop*. IEEE, 1994, pp. 279–288.

[13] S. Kishore and S. B Gangashetty, "Online text-independent speaker verification system using autoassociative neural network models," *Neural Networks*, 2001.

[14] G. S. Sivaram, S. Thomas, and H. Hermansky, "Mixture of auto-associative neural networks for speaker verification." in *Proc. InterSpeech*, 2011, pp. 2381–2384.

[15] Y. Konig, L. Heck, M. Weintraub, and K. Sonmez, "Nonlinear discriminant feature extraction for robust text-independent speaker recognition," in *Proc. RLA2C, ESCA workshop on Speaker Recognition and its Commercial and Forensic Applications*, 1998, pp. 72–75.

[16] T. Fu, Y. Qian, Y. Liu, and K. Yu, "Tandem deep features for text-dependent speaker verification," in *Proc. InterSpeech*, 2014.

[17] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. ICASSP*. IEEE, 2014, pp. 4052–4056.

[18] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, no. 4, pp. 788–798, 2011.

[19] N. Scheffer and Y. Lei, "Content matching for short duration speaker recognition," in *Proc. InterSpeech*, 2014.

[20] 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*. IEEE, 2014, pp. 1695–1699.

[21] L. Yuan, F. Tianfan, F. Yuchen, Q. Yanmin, and Y. Kai, "Speaker verification with deep features," in *IEEE WCCI 2014 Final Program and Book of Abstracts*, Beijing, China, July 2014.

[22] B. Scholkopft and K.-R. Mullert, "Fisher discriminant analysis with kernels," *Neural networks for signal processing IX*, 1999.

[23] M. McLaren and D. Van Leeuwen, "Source-normalised-and-weighted lda for robust speaker recognition using i-vectors," in *Proc. ICASSP*. IEEE, 2011, pp. 5456–5459.

[24] P. Matejka, O. Glembek, F. Castaldo, M. J. Alam, O. Plchot, P. Kenny, L. Burget, and J. Cernocky, "Full-covariance ubm and heavy-tailed plda in i-vector speaker verification," in *Proc. ICASSP*. IEEE, 2011, pp. 4828–4831.

[25] P. Kenny, T. Stafylakis, P. Ouellet, M. J. Alam, and P. Dumouchel, "Plda for speaker verification with utterances of arbitrary duration," in *Proc. ICASSP*. IEEE, 2013, pp. 7649–7653.

[26] Q. Jin and A. Waibel, "Application of lda to speaker recognition." in *Proc. InterSpeech*, 2000, pp. 250–253.

[27] C. Barras and J.-L. Gauvain, "Feature and score normalization for speaker verification of cellular data," in *Proc. ICASSP*, vol. 2. IEEE, 2003, pp. II–49.