Speaker Verification Using Deep Neural Networks: A Review

Amna Irum and Ahmad Salman

***Abstract*—Speaker verification involves examining the speech signal to authenticate the claim of a speaker as true or false. Deep neural networks are one of the successful implementation of complex non-linear models to learn unique and invariant features of data. They have been employed in speech recognition tasks and have shown their potential to be used for speaker recognition also. In this study, we investigate and review Deep Neural Network (DNN) techniques used in speaker verification systems. DNN are used from extracting features to complete end-to-end system for speaker verification. They are generally used to extract speaker-specific representations, for which the network is trained using speaker data in training phase. Speaker representation depends on the type of the model, the representation level, and the model training loss. Usually deep learning is crux of attention in computer vision community for various tasks and we believe that a comprehensive review of current state-of-the-art in deep learning for speaker verification summarize the utilization of these approaches for readers in speech processing community.**

***Index Terms*—Feature extraction, bottleneck features, deep features, end-to-end systems.**

I. Introduction

Speaker's voice recognition or simply speaker recognition is the process of recognizing the identity the speaker through the speech he/she utters. Speaker recognition system works on the principle that every speaker's voice is unique like finger prints thus can be used to identify the speaker or authenticate his/her claim. These systems in general analyze the characteristics or features in the speech which are different among speakers and are used in applications for speaker authentication, surveillance and forensics. Depending upon the applications, speaker recognition can be broadly divided into three categories i.e., speaker identification [1], speaker verification [2] and speaker diarization [3]. Speaker verification analyzes the speech to check whether the claimed speaker is genuine or impostor. Speaker's speech is compared with template speech patterns of many speakers already enrolled in the system. Hence, this is a process to check the authenticity of the speaker. Speaker identification process the speech signal to identify the speaker out of the pool of many speakers. This is a process of finding which speaker provides a given utterance in the speaker database. Speaker diarization is the process of segmenting the input speech signal according to speaker identity. Speaker segmentation and clustering comes under this same category. The methodologies used for any application of speaker recognition can be adopted for the other with trivial modification in the working of the system.

Generally, speaker verification consists of training, enrollment, and evaluation phases [2]. In training phase, the system is trained using the available data to learn the speaker-specific features from speech signals. In enrollment phase the speaker utterances are fed to trained system to get the speaker models and finally in evaluation, test speaker utterance model is created and compared with the already existed models, to see similarity with already registered speakers.

There are two types of speaker verification systems based on the type of data used for enrollment and recognition; they are text-dependent and text-independent mode of operations. In text-dependent recognition systems, speaker's speech text is kept same for enrolment as well as evaluation. Text-independent systems take different speech text for enrolment and evaluation; these types of systems are mostly used for speaker identification and verification.

In this paper, we have discussed the usage of Deep Neural Network (DNN) in speaker verification. DNN are used from extracting features to complete end-to-end system for speaker verification task. DNN is generally employed to extract speaker-specific representations, for which DNN is trained using speaker's data in training phase. Speaker representation depends on the type of the model, the representation level, and the loss function used in training. DNN consists of several hidden layers [4]. The generic set of equation governing any DNN are the relationship between input and hidden layer, it activation and output layer,

$$\boldsymbol{f}_l = W^T \boldsymbol{x} + \boldsymbol{b}_l \tag{1}$$

where $\boldsymbol{f}_l$ is the output of the l^{th} layer for $l = 1,2,3, \dots, L$, W is the weight matrix connecting input speech feature vector $\boldsymbol{x}$ with that hidden layer. Vector $\boldsymbol{b}_l$ is the bias vector of l^{th} layer. Depending upon application, different types of activation function can be applied to the function $\boldsymbol{f}_l$ at hidden layer neuron i.e., sigmoid activation $\sigma(\boldsymbol{f}) = 1/(1 + e^{-f})$or ReLU activation $r(\boldsymbol{f}) = \max(0, \boldsymbol{f})$, $\boldsymbol{f}(\boldsymbol{x}) = \max(0, \boldsymbol{x})$. Generally soft-max loss or cross entropy loss is used to train the deep network and is applied to the L^{th} layer as the output of the network.

$$o(\boldsymbol{f}_L) = -\log \frac{e^{f_{L-1}}}{\sum e^{f_{L-1}}} \tag{2}$$

Depending upon the type of deep network, the dimension of the hidden layers $\boldsymbol{f}_l$ varies, which represents the number

Manuscript received May 23, 2018; revised October 14, 2018.

The authors are with School of Electrical Engineering and Computer Sciences (SEECS), National University of Sciences and Technology (NUST), Islamabad, Pakistan (e-mail: amna.irum@seecs.edu.pk, ahmad.salman@seecs.edu.pk).

doi: 10.18178/ijmlc.2019.9.1.760

of neurons in any layer. The network is trained using error back propagation gradient descent algorithm. Equation (1) takes a different shape for different network type being fully-connected, time-delay or convolution neural network.

The organization of this paper is as follows: In section II we will describe the baseline system with survey of deep networks for speaker verification. Discussion and conclusion is presented in Section III.

II. Base Line System

The baseline system used for speaker verification is GMM-UBM system.

A. GMM-UBM and i-Vector Based System

i-vector approach has shown considerable improvement in speaker verification [5]. It consists of three sequential steps 1) extracting information/statistics from data 2) calculating i-vectors 3) then applying a probabilistic linear discriminant analysis (PLDA) on back end.

In a first step, universal background model (UBM) is created using sequence of feature vectors (e.g., mel-frequency-cepstral coefficients (MFCC). A UBM is Gaussian mixture model (GMM) trained on entire pool of speakers' speech data [6]. The trained background model is then used to calculate the speaker-specific model by adjusting its means according to the individual speaker's data. The UBM is represented as $\lambda_{ubm} = \{w_i, \mu_i, \sum_i\}\, C_i$ where, $w_i, \mu_i, and \sum_i$ are weights, mean and covariance of i^{th} component in C of Gaussian components respectively. During enrollment phase the S number of speakers are adapted according to UBM created from training data to create speaker models $\{\lambda_1, \lambda_2, \ldots, \lambda_s\}$. The GMM is specified as

$$p(\boldsymbol{x}|\lambda) = \sum_{c=1}^{C} w_c p_c(\boldsymbol{x}). \quad (3)$$

$p_c(x)$ is represented as

$$p_c(\boldsymbol{x}) = \frac{1}{(2\pi)^{d/2}|\sum_c|} exp\left(-\frac{(\boldsymbol{x}-\mu_c)^T \Sigma_c^{-1}(\boldsymbol{x}-\mu_c)}{2}\right). \quad (4)$$

where d the dimension of input vector $\boldsymbol{x}$. GMM parameters are estimated using expectation maximization algorithm [6]. In evaluation phase, likelihood ratio between the target model and background is calculated.

i-vector is low dimensional feature vector containing speaker-specific and speech variability information in a speech segment, which is the GMM super-vector by a single variability space. It is written as

$$M = \boldsymbol{\mu} + T\boldsymbol{\omega} \quad (5)$$

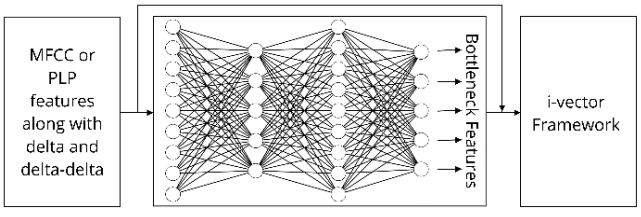

Fig. 1. Deep bottleneck features used for GMM-UBM/ i-vector.

where μ is mean super-vector, T is low rank total variability matrix and $\boldsymbol{\omega}$ is a low rank vector referred to as i-vector. PLDA model is then employed to generate verification scores by comparing i-vectors from different utterances [7]. Any model that can provide posteriors of K-classes per frame other than GMM can be used for i-vector calculation.

B. Features Extracted from DNN Used by GMM-UBM or i-Vector Based System

In these types of systems, DNN is used to extract frame by frame speaker information and calculate its utterance-level information. Output from the DNN is converted into i-vectors. PLDA is then used to generate the verification score [8], [9]. The DNN features are either used alone or combined with the conventional features (PLP or MFCC).

These systems are mostly used for text dependent speaker verification. They make use of DNNs which are trained for speech related tasks because GMM have shown inability to predict phonetic content in text-dependent speaker recognition. In [10], Lei et al uses Phonetically-Aware DNN to model speakers replacing GMM-UBM. The DNN calculates posteriors for tied-tri-phone state classes determined by a standard Automatic Speech Recognition (ASR). After calculating the posteriors, the zeroth and first order statistics are computed and sent to i-vector and PLDA back end. Similarly, [11] used DNN trained for speech recognition for speaker recognition and language recognition tasks. However, they extracted both, the bottleneck features from DNN bottleneck layer and DNN posteriors from the last layer of the network. Features extracted from bottleneck layer are used to create the GMM-UBM model followed by i-vector framework. However, the posteriors are sent to i-vector frame directly. Both frameworks and their combined variant outperformed baseline i-vector system.

Yuan Liu *et al*. [12] make use of large amount of data available for speech recognition without speaker labels for both text-dependent and text-independent speaker verification tasks. Unsupervised training is done on DNN in the form of Restricted Boltzmann Machines (RBM) utilizing unlabeled ASR data. Trained network grabs speaker and speech related information. The output layer of DNN gives more discrimination on basis of speech. To extract the speaker-specific information as well, features are grabbed from the middle layer, called bottleneck features. Bottleneck features along with MFCC and PLP are used to create GMM-UBM followed by i-vector calculations and PLDA scoring. Experiments are done for both text-dependent and text-independent speaker verification tasks.

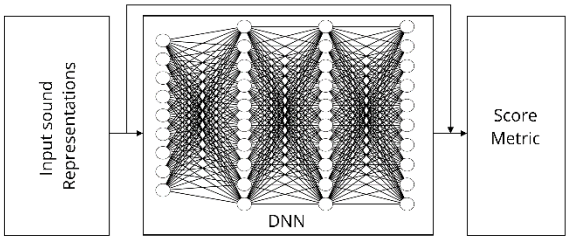

Fig. 2. Deep features systems used for speaker verification.

Bottleneck features along with MFCCs are also concatenated together [13] and GMM/UBM is created. Bottlenecked features are extracted from senone-discriminate

deep neural network.

In another work, cascade of two bottleneck features from two DNN are termed as Stacked Bottleneck Features [9]. The output of the first DNN is stacked in time, defining context-dependent input features for the second DNN. The bottleneck features along with the MFCCs are analyzed using GMM-UBM i-vector based systems.

In [14], Baum-Welch statistics are calculated by combining the posteriors from DNN and 60-dimensional MFCCs. Posteriors grab acoustic phonetic information about speaker and speech phonetic sounds are used for text-independent speaker recognition. To learn the speech information for longer utterances, input speech frames along with neighboring frames are used. The combined features are then used for i-vector creation. As back end classifier, a generative PLDA model is used.

C. Features Extracted from DNN Used as Posteriors

These systems use DNN to extract the speaker related information/features and apply the similarity metric to check the scores of verifications. The extracted features are either taken from some low dimensional layer called bottleneck layer [2], [10], [15] or last layers of DNN i.e., DNN posteriors. Outputs from bottleneck layer contain the frame level information, which was aggregated before score checking.

TABLE I: COMPARATIVE STUDY OF VARIOUS SPEAKER VERIFICATION SYSTEM BASED ON DNN ARCHITECTURES

Reference	Type of System	Input Features	DNN Type	Score Function	Baseline System	Dataset	Score (%EER)
[10]	Text-dependent	40 log Mel-filter bank coefficients	7-layered, fully-connected	PLDA	UBM/i-vector	NIST SRE'12, Noisy narrowband	1.39
[12]	Text-independent	39 dimension PLP	7-layered RBM	Cosine Distance	GMM-UBM	NIST SRE'05-06	0.88
[14]	Text-independent	60 dimension MFCCs	5-layered	PLDA	UBM/i-vector	NIST SRE'12, Switchboard I, II, III	1.58
[16]	Text-dependent	39 dimension PLP	4-layered, fully-connected	Cosine Distance	UBM/i-vector	Self-created	1.21
[17]	Text-dependent	20 dimension MFCCs	4-layeres, fully-connected	PLDA	UBM/i-vector, GMM-DTW	RSR 2015 Specifically designed for text dependent speaker verification.	0.2
[18]	Text-dependent	39 dimension PLP	4-layered, fully-connected	PLDA	GMM-UBM, d-vector, j-vector	SR 2015	0.54
[19]	Text-dependent	40 dimension MFCCs	7-layered, multi-splice time delay	GPLDA	GMM-UBM	NIST SRE'10	7.2
[20]	Text-independent	Phone-blind & Phone-aware 40 dimensional d-vectors	7-layered, time-delay	PLDA	UBM/i-vector	Fisher dataset, CSLT-CUDGT2014	8.37
[21]	Text-independent	20 dimension MFCCs	4-layered, temporal pooling	PLDA	UBM/i-vector	US English telephonic speech,	5.3

In [16], DNN is trained to make a background model from the frame level features using supervised learning. Trained DNN is able to extract the speaker-specific features and speaker model is computed by averaging, per frame, last layer's output of DNN for every speaker in enrollment data. These average vectors are called d-vectors. During testing phase, these speaker model d-vectors are compared with test speaker d-vectors. Although i-vector framework outperformed the d-vectors, the fusion of i-vector and d-vectors results have shown better performance as compared to i-vector alone. d-vectors are found to be robust against noise.

To exploit text information fully in text-dependent speaker verification, [17] used DNN trained for ASR to extract posteriors. To incorporate the sequence information while modeling posteriors, Dynamic Time Wrapping (DTW) is used. DTW compares the posteriors of two utterances and shown considerable improvement as compared to i-vector. DNN posteriors are called j-vectors by [18] where single DNN framework is trained to learn the text content and speaker related features simultaneously. These jointly learned features then applied to PLDA with classes defined as multi-task labels on both speaker and text as the decision function.

Short utterances of around 15 seconds for speaker verification are used by [19] where they train DNN-senone using speech data. The posteriors are then sent to for Gaussian probabilistic linear discriminant analysis (GPLDA).

DNN posteriors extracted from deep network, composed of convolutional layers followed by time delayed layers, are employed for speaker verification in multilingual setting in [20]. In multilingual system, speaker can speak in different language at enrollment and testing time. Other systems cannot perform well in this scenario because GMM-UBM and i-vector based systems model speakers using their speech characteristics along with phonetic information. When this information, which is language dependent is omitted, these systems compromise on their performance. Convolutional layers are followed by time delayed layers used for grabbing

temporal context. This system doesn't take into account the language while extracting features. Language related information can be added by modeling any model that can discriminate phone.

D. End-to-End Systems

An end-to-end system treats the entire system as a whole adaptable black box. The process of feature extraction and classifier training are achieved together with an objective function that is consistent with the evaluation metric. NIN (network in network) layers are used to construct the end-to-end system. In [21], the system consists of four NIN layers followed by temporal pooling layer. Temporal pooling layer aggregate the variable length frames and send the data to hidden NIN layer followed by linear pooling layer. The output of the network is called speaker embedding. These embedding are compared using distance metric based on objective function and have shown considerable performance as compare to i-vector. Further work [22] used DNN for extracting the speaker discriminative property and used PLDA back end for scoring. Deep network in this approach is composed of three parts i.e., first part consists of five layers arranged as in [21] to capture temporal dependencies in speech signal to extract temporal context, followed by statistical pooling layer which took the input from the output of the layer below. It sums the input segment and calculated the mean and standard deviation. The segment-level output from the pooling layer is passed to additional hidden layers and then to final softmax output layer. Hidden layer outputs are called embedding. The DNN is trained using variable length speech segments to classify speakers using multiclass entropy function. The embeddings after applying the dimensionality reduction are compared using PLDA.

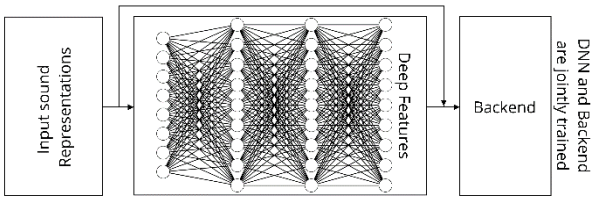

Fig. 3. End-to-end system.

The comparisons between various DNN based approaches for speaker verification are summarized in Table I.

III. Discussion and Conclusion

It is well known fact that speech carries a mixture of information that includes, message, speaker's emotion information and speaker-specific information [4]. Speaker-specific information is utilized in tasks like speaker identification and speaker verification where aim for any technique is to extract and utilize good quality features representing intrinsic characteristics of speaker's vocal apparatus. There are several approaches presented so far for speaker recognition, which includes generative modelling like GMM [1], kernel-based discriminative learning [23] or enhancement in decision-making approaches [24]. All these approaches contributed towards achieving better scores on speaker recognition related tasks. However, the problem of extracting unique and invariant-to-corruption speaker-specific features for favorable results is still an open topic of research as mixing of speaker-specific information with other information components of speech makes speaker recognition systems to compromise on their performance [25]. In addition, the influence of inter-speaker similarity, intra-speaker dissimilarity, channel variability and additive noise exacerbates the situation [26]. Therefore, the need of getting pure features that are insensitive to these corruptions is inevitable. In some studies like [27], [28], efforts are made to extract speaker-specific information from speech signal with an assumption that message or linguistic information is dominant information component and can be sifted out easily. This was achieved by a mapping of speech representation e.g., MFCC or PLPC to speaker-specific information by suppressing linguistic component with prior assumption that it occupies lower frequency sub-bands in speech spectrum while speaker-related component is reflected by higher frequency bands.

Currently, deep learning approaches are among state-of-the-art in achieving promising results in computer vision and speech information processing [29], [30]. Deep architectures are being used from speech recognition to speaker recognition where aim is to employ highly nonlinear and complex parametric mathematical model in supervised and unsupervised learning paradigm to estimate better speaker characteristics implicitly in output space without any prior assumption about the explicit segregation of information components as in [27] and [28]. Conventional machine learning algorithms lack the required depth and nonlinearity to accurately map speaker model in the output space. These are generally called shallow architectures with examples including but not limited to Support Vector Machines (SVM) [23], Multilayer Perceptrons (MLP) [31], Nearest Neighbor Classifiers [32], Principal Component Analysis (PCA) and Independent Component Analysis (ICA) [27]. Previously, these approaches are used in speaker recognition tasks but fail to produce reasonable results on large and highly variable datasets. In contrast, given adequate data, DNNs are trained to approximate unique task-specific information through multiple layers of hierarchical learning with each layer realizing nonlinear mathematical mapping from the preceding layer. Using appropriate loss function for DNN training, channel and noise distortion are implicitly suppressed making output feature more suitable for classification.

In their paper, we have reviewed various approaches that are based on deep neural networks for the task of speaker verification. There are three types of DNN-based speaker verification approaches that are discussed 1) Bottleneck features from DNN+MFCC or PLP and i-vector back end 2) d-vectors embeddings learned from DNN and used for scoring 3) End-to-end system. These three approaches are different in many ways. In training phase, the bottleneck features are trained separately in supervised or unsupervised fashion and combined with other data to be used for back end systems for speaker verification. Similarly, the d-vectors, which act as a front end, are trained separately for grabbing features. While the end-to-end system front end and back end are jointly trained.

The training objective of first two approaches is to learn the features that discriminate speakers. But the end-to-end

system is trained pair-wise to find whether the utterances are from same speaker or different speaker. Quality and quantity of pair of utterances are important for end-to-end system.

According to [21], if trained on small data set, end-to-end system does not perform well but for larger data set it outperforms d-vectors which is consistent with the fact for better generalization the end-to-end DNN system requires large set of training examples otherwise it usually results in overfitting to the training data while produce unfavorable performance on unknown test data [33]. On large dataset, the gain of end-to-end system is due to the fact that the system is using utterance-level input which better captures speech-dependent speaker's statistics from variable length speech segments. In contrast, results reported in [22] show that end-to-end system performs worse as compared to d-vector based systems and i-vector based systems on short training utterances. Their results are consistent with [21].

IV. Future Work

All above mention approaches have potential for improvement and can contribute towards better speaker verification system. Bottleneck feature based systems have shown considerable improvements and provide better speaker-specific information as compared to input raw features. Generally, input features used for calculation of bottleneck features are MFCCs or PLPs. Future work should consider all the possible speech representations as inputs and their combination to generate the bottleneck features that are more speaker-specific. If different deep architectures and loss functions are applied to these systems individually, each system has potential to outperform their present performances. The bottleneck features and embedding learned by first two approaches have the potential to be used for applications other than speaker verification [34]-[36]. Convolutional neural networks (CNNs) have been applied for feature extraction and have shown reasonable results. In future, the advantages of all these systems should be carefully examined and combined to make more efficient speaker verification system. Furthermore, there is a great need to explore the possibility of optimum deep architecture with extensive study to determine a correct number of network layers with reasonable choice of encoding scheme and type of nonlinear activation functions. This is important due to the fact that unnecessary network neurons not only burden the training due to increase in computational complexity and difficulty in loss convergence but they also generate spurious data which eventually acts as noise and corrupt the output features. Deep networks are successful in getting state-of-the-art performance especially in computer vision tasks including generic object recognition, object localization and classification [37], facial recognition [38] and optical character recognition [39] from the images. However, the information of interest in this case is already predominant and relatively easy to capture and learn due to clearer shape, texture and color information. In contrast, speaker-specific information is minor information in speech signals, dominated by linguistic information component. Therefore, the choice of carefully tailored deep network with reasonable parameters to extract subsided yet important information is inevitable for better performance of speaker recognition systems. The Network loss function plays critical role in adapting to the correct mapping of input speech representation, which is a mixture of various information components, to output features, which should be highly enriched with speaker-specific information. Therefore, various options of loss functions including triplet losses [40], statistical losses [33], [41] and maximum margin losses [42] along with their combination are on table for empirical study. It is also important to analyze the effect of input data size, its variation in terms of channel effects and noise distortion and consequently its influence on the choice of deep network architecture. Similarly, further investigation needs to be done to check the performance of these systems when training size increases.

References

[1] A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19-41, 2000.

[2] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. on Speech and Audio Processing*, vol. 3, pp. 72-83, 1995

[3] S. Tranter and D. Reynolds, "An overview of automatic speaker diarisation systems," *IEEE Trans. Audio Speech Lang. Process*., vol. 14, no. 5, pp. 1557-1565, 2006.

[4] A. Salman and K. Chen, "Exploring speaker-specific characteristics with deep learning," in *Proc. IJCNN*, pp. 103-110, 2011.

[5] T. Yamada, L. Wang, and A. Kai, "Improvement of distant-talking speaker identification using bottleneck features of dnn," in *Proc. INTERSPEECH*, pp. 3661–3664, 2013.

[6] Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. G. Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. ICASSP*, pp. 4080–4084, 2014.

[7] 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*, pp. 1714–1718, 2014.

[8] P. Matejka *et al*., "Analysis of dnn approaches to speaker identification," in *Proc. ICASSP*, pp. 5100-5104, 2016.

[9] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to-end neural speaker embedding system," *arXiv preprint*, arXiv:1705.02304, 2017.

[10] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Proc. ICASSP*, pp. 5115-5119, 2016.

[11] F. Richardson, D. Reynolds, and N. Dehak, "A unified deep neural network for speaker and language recognition," *arXiv preprint*, arXiv:1504.00923, 2015.

[12] S. David, G. Pegah, P. Daniel, G. R. Daniel, C. Yishay, and K. Sanjeev, "Neural network-based speaker embeddings for end-to-end speaker verification," *IEEE Spoken Language Technology Workshop (SLT)*, 2016.

[13] M. J. Alam, P. Kenny, and V. Gupta, "Tandem features for text-dependent speaker verification on the reddots corpus," in *Proc. INTERSPEECH*, pp. 420–424, 2016.

[14] 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.

[15] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio Speech Lang. Process*., vol. 19, pp. 788–798, May 2010.

[16] Ehsan, L. Xin, M. Erik, L. M. Ignacio, and G.-D. Javier, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. ICASSP*, pp. 4080–4084, 2014.

[17] S. Dey, S. Madikeri, M. Ferras, and P. Motlicek, "Deep neural network based posteriors for text-dependent speaker verification," in *Proc. ICASSP*, pp. 5050–5054, 2016.

[18] N. Chen, Y. Qian, and K. Yu, "Multi-task learning for text-dependent speaker verification," in *Proc. INTERSPEECH*, pp. 185-189, 2015.

[19] A. Kanagasundaram, D. Dean, S. Sridharan, and C. Fookes, "DNN based speaker recognition on short utterances," *arXiv preprint*, arXiv:1610.03190, 2016.

[20] L. Li, D. Wang, A. Rozi, and T. Fang, "Cross-lingual speaker verification with deep feature learning," *arXiv preprint*, arXiv:1706.07861, 2017.

[21] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," *IEEE Spoken Language Technology Workshop (SLT)*, 2016.
[22] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Proc. INTERSPEECH*, pp. 999-1003, 2017.
[23] W. Campbell, J. Campbell, D. Reynolds, E. Singer, and P. Torres-Carrasquillo, "Support vector machines for speaker and language recognition," *Comput. Speech Lang*., vol. 20, pp. 210-299, 2006.
[24] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Trans. Audio, Speech and Lang. Processing*, vol. 15, pp. 1435-1447, 2007.
[25] M. Joos, "Acoustic phonetics," *Language*, vol. 24, pp. 1-136, 1948.
[26] R. Turner, T. Walters, J. Monaghan, and R. Patterson, "Statistical formant-pattern model for estimating vocal tract length from formant frequency data," *Journal of Acoust. Soc. America*, vol. 125, pp. 2374-2386, 2009.
[27] Jang, T. Lee, and Y. Oh, "Learning statistically efficient feature for speaker recognition," in *Proc. ICASSP*, pp. 437-440, 2001.
[28] N. Malayath, N. Hermansky, S. Kajarekar, and B. Yegnanarayana, "Data-driven temporal filters and alternatives to GMM in speaker verification," *Digital Signal Process*, vol. 10, pp. 55-74, 2000.
[29] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, pp. 1-127, 2009.
[30] Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring strategies for training deep neural networks," *J. Machine Learning Research*, vol. 17, pp. 1-40, 2009.
[31] D. Rodriguez-Porcheron and M. Faundez-Zanuy, "Speaker recognition with MLP classifier and LPCC codebook," in *Proc ICASSP*, pp. 1005-1009, 1999.
[32] Kacur, R. Vargic, and P. Mulinka, "Speaker identification by K-nearest neighbours: Application of PCA and LDA prior to KNN," in *Proc. Int. Conf. Systems, Signals and Image Processing*, 2011.
[33] Chen and A. Salman, "Learning speaker-specific characteristics with a deep neural architecture," *IEEE Trans. Neural Networks*, vol. 22, pp. 1744-1756, 2011.
[34] A. A. Mallouh, Z. Qawaqneh, and B. D. Barkana, "New transformed features generated by deep bottleneck extractor and a GMM–UBM classifier for speaker age and gender classification," in *Proc. Neural Computing and Applications*, pp. 1-13, 2017.
[35] S. Petridis and M. Pantic, "Deep complementary bottleneck features for visual speech recognition," in *Proc. ICASSP*, pp. 2304-2308, 2016.
[36] A. Torfi N. M. Nasrabadi, and J. Dawson, "Text-independent speaker verification using 3D convolutional neural networks," *arXiv preprint*, arXiv:1705.09422, 2017.
[37] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. CVPR*, pp. 779-788, 2016.
[38] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc BMVC*, pp. 1-12, 2015.
[39] R. Salakhutdinov and G. Hinton, "Learning a nonlinear embedding by preserving class neighbourhood structure," in *Proc. AISTATS*, 2007.
[40] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," *Int. Workshop on Similarity-Based Pattern Recog.*, Springer, Cham, 2015.
[41] C. A. Salman, "Extracting speaker-specific information with a regularized Siamese deep network," in *Proc. NIPS*, pp. 1-9, 2012.
[42] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively with application to face verification," in *Proc. CVPR*, 2005.

Amna Irum is PhD student working in area of speaker verification using deep learning student in at Department of Computing, School of Electrical Engineering and Computer Science, National University of Sciences and Technology, Islamabad, Pakistan.

Ahmad Salman is associated with School of Electrical Engineering and Computer Science, National University of Sciences and Technology, Islamabad, Pakistan. He did his PhD from University of Manchester, UK in 2012 in computer science and specializes in machine learning and speech information processing.