



Analysis of Deep Generative Model Impact on Feature Extraction and Dimension Reduction for Short Utterance Text-Independent Speaker Verification

Aref Farhadipour¹ · Hadi Veisi²

Received: 22 July 2023 / Revised: 17 March 2024 / Accepted: 18 March 2024
© The Author(s) 2024

Abstract

Speaker verification is a biometric-based method for individual authentication. However, there are still several challenging problems in achieving high performance in short utterance text-independent conditions, maybe for weak speaker-specific features. Recently, deep learning algorithms have been used extensively in speech processing. This manuscript uses a deep belief network (DBN) as a deep generative method for feature extraction in speaker verification systems. This study aims to show the impact of using the proposed method in various challenging issues, including short utterances, text independence, language variation, and large-scale speaker verification. The proposed DBN uses MFCC as input and tries to extract more efficient features. This new representation of speaker information is evaluated in two popular speaker verification systems: GMM-UBM and i-vector-PLDA methods. The results show that, for the i-vector-PLDA system, the proposed feature decreases the EER considerably from 15.24 to 10.97%. In another experiment, DBN is used to reduce feature dimension and achieves significant results in decreasing computational time and increasing system response speed. In a case study, all the evaluations are performed for 1270 speakers of the NIST SRE2008 dataset. We show deep belief networks can be used in state-of-the-art acoustic modeling methods and more challenging datasets.

Keywords Speaker verification · Deep belief networks · i-vector · GMM · Feature extraction

✉ Aref Farhadipour
aref.farhadipour@uzh.ch

Hadi Veisi
h.veisi@ut.ac.ir

¹ Department of Computational Linguistics, University of Zurich, Andreasstrasse 15, Zurich, Switzerland

² Faculty of New Sciences and Technologies, University of Tehran, North Kargar, Tehran, Iran

1 Introduction

In the speaker verification task, a person claims an identity, and the system attempts to accept or reject it using the features of the individual's speech. There are two text-dependent and -independent scenarios considering whether the text expressed in the two training and testing sections is identical or different, respectively. In the first one, since speech content can be multifarious, systems are faced with significant variations in learning and modeling individuals' speech specifications. Therefore, designing a robust and efficient text-independent speaker verification system is challenging.

The general parts of a speaker verification system include pre-processing, feature extraction, acoustic modeling, and decision making. The pre-processing step tries to form all the signals in a single format with several tasks like removing noise and silence. Due to its sampling frequency, each speech segment has a large number of samples that are not particularly useful for speaker verification, and this increases the computational time and complicates the system. Consequently, the feature extraction stage tries to extract valuable and low-dimensional coefficients, which describe the speech signal. One of the most commonly used features is the Mel Frequency Cepstral Coefficient (MFCC).

Based on extracted features, the acoustic modeling stage endeavors to create specific models for each one of the speakers via a modeling algorithm like the Gaussian Mixture Model-Universal Background Model (GMM-UBM) [43]. Finally, the decision is made by comparing the created model and the test utterance feature.

The i-vector approach aims to identify a specific speaker and channel information based on a fixed-length identity vector (i-vector). Several experiments have been conducted to extract these i-vectors from GMM [11] or deep neural networks (DNN) [3, 44, 53]. Although deep learning-based techniques like x-vector [12, 28, 51], in which the averaged activations of the last hidden layer of a deep neural network are selected as the identity vector, or end-to-end architecture [13, 34, 52] are used for speaker recognition, many modern speaker verification systems are still based on the i-vector [7, 8, 39].

Speaker recognition systems encounter significant challenges, notably data scarcity [2] and short-duration speech [41], which impact their design and performance. Our research focuses on developing specialized solutions that effectively address the unique difficulties posed by short utterances to improve the overall effectiveness of the speaker verification system. In a long-duration speech, which is longer than 30 s, the i-vector and the PLDA-based systems perform well; however, low-level performance is expected for short utterances [10]. In identical conditions, the i-vector extracted from the short utterances has more intra-class variations than long utterances [42]. It should be mentioned that there have been various efforts to address the issue as mentioned above. One of the ideas was the improvement of modeling of the variations in the i-vector extracted from these short-length statements [9]. Kanagasundaram et al. [27] proposed the normalization and variance modeling of utterances at the i-vector level. Moreover, the phonetic information is associated with the acoustic modeling. Several studies have attempted content matching through phonetic details [54].

The session variability vectors were also used to estimate the phonetic components instead of the i-vector extracted from an utterance. Some studies have focused on

the i-vector mapping from the short utterance i-vectors to the long version [19, 42]. Kheder et al. [33] trained a GMM with the short and long utterances to perform the i-vector mapping from the short to long versions. Instead of GMM-based mapping functions, nonlinear function-based mappings like the convolutional neural network (CNN) have received attention in recent years [46].

Deep neural networks improved the performance of speech recognition and speaker recognition systems. Takamizawa et al. [50] proposed a speaker identification system based on a deep neural network that identified whether or not the same speaker uttered two speech samples by focusing on the phonemes, which had very short durations. The architecture of the proposed system was based on ResNet. In recent years, several studies used CNN algorithms for acoustic modeling and feature extraction [14, 29, 38], and several research works have focused on speaker spoofing challenges [25, 49, 59].

Variational autoencoder (VAE) was used in speaker verification to improve environment mismatch between training and testing, such as noises and channel effects [56]. Evaluations on the NIST SRE2016 dataset showed 15.54% and 7.84% EER for Tagalog and Cantonese languages using i-vector and PLDA. In [55], a VAE was proposed to transform x-vectors into a regularized latent space. Experiments demonstrated that this VAE-adaptation approach transformed speaker embeddings to the target domain and achieved 12.73% EER for 77 speakers.

Some other works have also used deep belief networks (DBN) in speaker recognition. Ghahabi et al. [18] proposed an adaptation for a universal DBN as the background model for each speaker. Additionally, an impostor selection method was introduced to help the DBN outperform the cosine distance classifier. The evaluation was performed on the core test condition of the NIST SRE2006 corpora, and a 10% improvement in EER was reported. In [4], DBN was used as a feature extractor, and performance improvement was noted. This improvement was achieved by utilizing the spectrogram as input for DBN.

Feature extraction has considerably affected the performance of speaker verification systems. It has been argued that deep neural networks can model nonlinear functions [20]. This proposition creates the idea of effectively extracting speech features. The present study aims to investigate designing a short utterance text-independent speaker verification system using DBN in an autoencoder architecture to extract speakers' features. This DBN tries to improve the performance of the speaker verification systems by incorporating regular MFCC features into the network and extracting new feature sets in an unsupervised learning strategy. Moreover, an effort was made to reduce the feature vector dimensions to decrease the computational time using DBN.

The current study is organized as follows: Section 2 describes two GMM-UBM and i-vector-PLDA speaker verification systems. Section 3 discusses the proposed feature extractor and deep belief network theory. Section 4 presents the experimental settings and criteria. The simulation results and the analysis of the proposed systems are presented in Sect. 5. Finally, the conclusion is presented in Section 6.

2 Proposed Speaker Verification Systems

2.1 GMM-UBM Speaker Verification

GMM-UBM was the most commonly used method in the classical speaker verification systems. Figure 1 shows the framework of the proposed GMM-UBM-based system. Accordingly, audio files are pre-processed and converted to efficient features in the front-end block. The output of the front-end block, MFCC features, is the input for the proposed DBN block. The UBM has an essential role in GMM-UBM-based systems. It is a GMM-based background model built from the speech samples of the nontarget speakers in the development phase. The purpose of the UBM is to achieve a speaker-independent distribution, covering all the probabilistic space, so the feature distribution of a particular speaker can be extracted from it. More diversity in speakers, channels, and vocabularies can make a more general UBM. The expectation–maximization (EM) algorithm is commonly utilized to construct this model [37].

In the Enrollment phase, every speaker's train utterances are applied to the UBM, and according to each speaker's information, the background model parameters, including averages, variances, and coefficients, are updated to make the acoustic model of each person. This update is accomplished using the MAP algorithm [17]. During the verification phase, there are two assumptions of H_0 and H_1 for each utterance U claims the speaker's identity S .

H_0 : if U belonged to the speaker S .

H_1 : if U does not belong to the speaker S .

These two assumptions are examined using the two specific speaker and background models. Finally, Eq. 1 makes the decision using the probability factor.

$$\Lambda = \frac{1}{L} \log \frac{p(U|H_0)}{p(U|H_1)} = \begin{cases} \geq 0 & \text{accept } H_0 \\ < 0 & \text{accept } H_1 \end{cases} \quad (1)$$

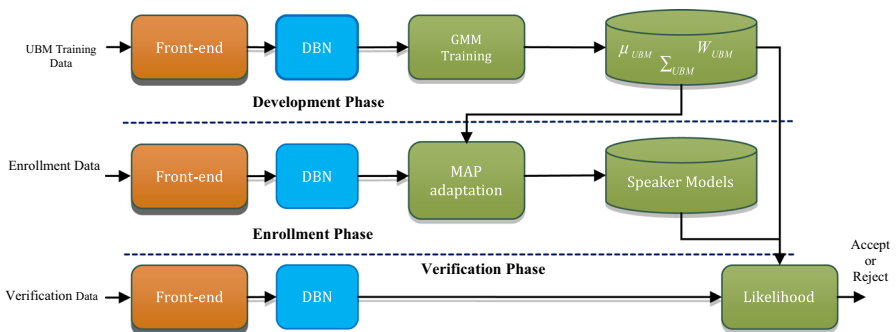


Fig. 1 Framework of the proposed GMM-UBM speaker verification system with DBN block

where $P(U|H_i), i = 0, 1$ is the conditional probability of the hypothesis H_i for utterance U . L denotes the number of frames per view of U . Generally, the UBM is utilized as the impostor model during the testing phase.

2.2 i-Vector Speaker Verification

Figure 2 shows the framework of the proposed i-vector system. The i-vector refers to the identity vector for each speaker. This model can process variable-length speech signals by mapping them to a fixed-length and low-dimensional vector. The i-vector extraction block tries to represent the mismatches like intra-speaker variations and the session variability in the GMM. The idea behind the i-vector is based on the assumption that the speaker-dependent and channel-dependent variations can be incorporated into a separate low-dimensional subspace through the joint factor analysis (JFA) technique. This algorithm eliminates or reduces intra-speaker changes and channel effects [30].

In the development phase, this system employs the UBM. The UBM and total variability models are trained as a space to represent the changes. In other words, each utterance can be represented as a supervector M , which is the mean vector of the total GMMs belonging to each speaker. This supervector is separately calculated for every speaker as follows:

$$M = m + Vy + Ux + Dz \tag{2}$$

where m is a supervector-like array derived from the UBM and assumed to be independent of the channel and the speaker information. The three parameters of $V, U,$ and D represent the characteristics of the speaker, subspace, and sessions, respectively. The two components y and x signify the speaker and channel components, respectively. The Dz is speaker’s residual information that is not included in Vy . In the Enrollment phase, the i-vector is extracted regarding each speaker. It is worth noting that the channel components also have the speaker information itself, so the subspace is proposed

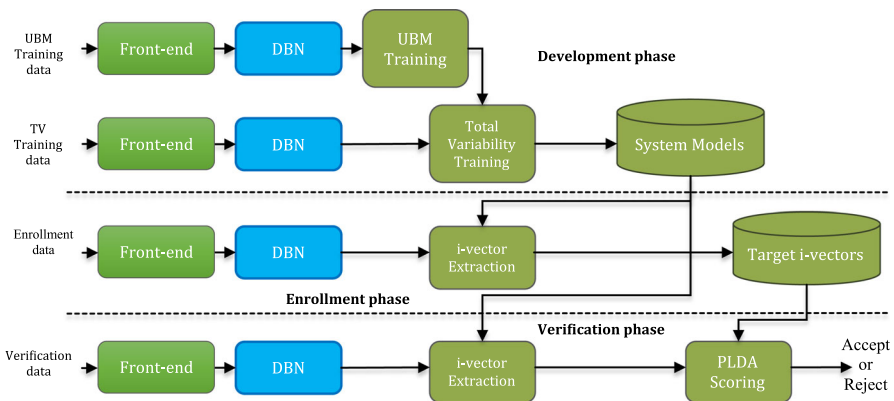


Fig. 2 Framework of the proposed i-vector-based speaker verification system with DBN block

for both variables [53]. This GMM-based supervector, depending on the speakers and sessions, is defined as follows:

$$M = m + Tw \quad (3)$$

where T is the matrix of the speaker variability of the sessions, and the component w denotes the identity vector or the i -vector. Baum–Welch statistical algorithm is used to train the full variable subspace, which is defined as follows:

$$N_c = \sum_t P(c|y_t, \Phi) \quad (4)$$

$$F_c = \sum_t P(c|y_t, \Phi) y_t \quad (5)$$

where N_c and F_c express zero- and first-order statistics, y_t is the feature sample at the time t , Φ denotes the UBM with the mixture component c ($c = 1, \dots, C$), which is the Gaussian index, and $P(c|y_t, \Omega)$ is the posterior probability of the mixture component c that produces the y_t vector. In the verification phase, these identity vectors can be used in several classifiers, including cosine similarity, LDA [36], and PLDA [31].

3 Feature Extraction Using Deep Generative Model

The feature extractor is one of the most important parts of a speech processing system that extracts low-dimensional and efficient information from the input speech signal. MFCC is one of the most popular features in speech processing. The MFCC algorithm aims to extract the envelope of speech signals. This feature is known as a short-term feature. On the other side, several long-term features have been introduced for different speech recognition scenarios [58], which may not properly describe the speaker's specific information despite the contextual information. Therefore, finding particular features that are effective in speaker recognition can significantly affect the performance of these systems.

Deep learning is a novel method widely used for feature extraction from raw data or classical feature engineering methods [48]. Accordingly, this research applies a special type of probabilistic generative deep neural network called deep belief networks [22] under autoencoder architecture that can be an unsupervised feature extractor. Recent studies on speaker verification have exploited the benefits of DNNs in acoustic modeling and feature extraction [48]. To use DNNs in acoustic modeling, the network must be trained with the specific information of each speaker.

The deployment of DNNs as the feature extractor consists of supervised [47] and unsupervised [16] scenarios. The supervised approach needs labeled data, and the selection of the labels is important and dependent on the context (text-independent or text-dependent). In the proposed unsupervised mode, the DBN is used in the autoencoder architecture, in which the network attempts to reconstruct the information of the input layer in the output layer.

DBNs are generative models made up of stacked restricted Boltzmann machines (RBM). Each RBM is a two-layer network that models the distribution of input (visible) layer data in the output (hidden) layer based on the weights of connections [24]. There are no visible–visible and hidden–hidden connections. In other words, The RBM is an energy-based model whose probability of joint distribution is based on its energy function as follows:

$$P(v, h|\lambda) = \frac{e^{-E(v,h)}}{Z} \tag{6}$$

The configuration energy (v, h) in this network is presented in Eq. (7):

$$E(v, h) = - \sum_{i=1}^V \sum_{j=1}^H w_{ij} v_i h_j - \sum_{i=1}^V a_i v_i - \sum_{j=1}^H b_j h_j \tag{7}$$

where v_i and h_j are the states of the visible unit i and the hidden unit j , respectively, w_{ij} is the weight between v_i and h_j , and a_i and b_j are the biases. Therefore, an expression for the marginal probability can be written by assigning an RBM to a visible vector v ,

$$P(v|h) = \frac{\sum_h e^{-E(v,h)}}{Z}. \tag{8}$$

where $Z = \sum_v \sum_h e^{-E(v,h)}$ is the normalizing constant. The derivative estimation of the log probability $P(v|\lambda)$ concerning the model parameters λ is as follows:

$$\frac{\partial \log P(v|\lambda)}{\partial w_{ij}} = \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}} \tag{9}$$

where $\langle \alpha \rangle_{\text{data}}$ and $\langle \alpha \rangle_{\text{model}}$ are the expectation of α estimated from the data and the model, respectively. The derivative in (9) leads to the following learning rule:

$$\Delta w_{ij} = \varepsilon (\langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}}) \tag{10}$$

where ε is the learning rate. The hidden neurons are conditionally independent, presenting the visible vector. Then, the binary state of each hidden unit h_j is set to one with the following probability:

$$P(h_j = 1|v) = \psi \left(\sum_i w_{ij} v_i + b_j \right) \tag{11}$$

where $\psi(\cdot)$ is the sigmoid logistic function. Likewise, Eq. (12) presents the visible binary neuron:

$$P(v_i = 1|h) = \psi \left(\sum_j w_{ij} h_j + a_i \right) \tag{12}$$

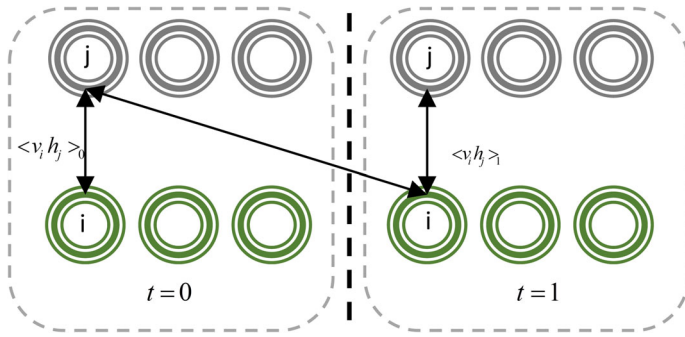


Fig. 3 One-step contrastive divergence of an RBM

The estimation of the input data $\langle v_i h_j \rangle_{\text{data}}$ is straightforward, but approximated methods such as contrastive divergence (CD) [21] are required to estimate the $\langle v_i h_j \rangle_{\text{model}}$ term.

The one-step CD of an RBM is shown in Fig. 3. The approximation for the gradient regarding the visible to hidden weights is as follows:

$$\begin{aligned} \Delta w_{ij} &= -\varepsilon(\langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\infty}) \\ &\approx -\varepsilon(\langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_1) \end{aligned} \tag{13}$$

where $\langle . \rangle_{\infty}$ denotes the expectation computed with the samples generated by running the Gibbs sampler in infinite steps, and $\langle . \rangle_1$ is the expectation for running in one step. Similarly, the learning rules for the bias parameters are as follows:

$$\begin{aligned} \Delta a &= -\varepsilon(\langle v \rangle_{\text{data}} - \langle v \rangle_1) \\ \Delta b &= -\varepsilon(\langle h \rangle_{\text{data}} - \langle h \rangle_1). \end{aligned} \tag{14}$$

When the visible unit v is real-valued like the MFCC vector and the hidden unit h is binary, the RBM energy function can be modified to enable it to adapt such variables, presenting a Gaussian–Bernoulli RBM (GRBM). The energy of GRBM is defined as follows [20]:

$$E(v, h) = \sum_{i=1}^V \frac{(v_i - a_i)^2}{2\sigma_i^2} - \sum_{i=1}^V \sum_{j=1}^H \frac{v_i}{\sigma_i} w_{ij} h_j - \sum_{j=1}^H b_j h_j \tag{15}$$

where the variance parameters σ^2 are commonly fixed to a predetermined value instead of being learned from training data. To train a GRBM using the CD algorithm, two conditional distributions for Gibbs sampling are derived as follows:

$$P(h_j = 1 | v) = \psi(b_j + \sum_i \frac{v_i}{\sigma_i} w_{ij}) \tag{16}$$

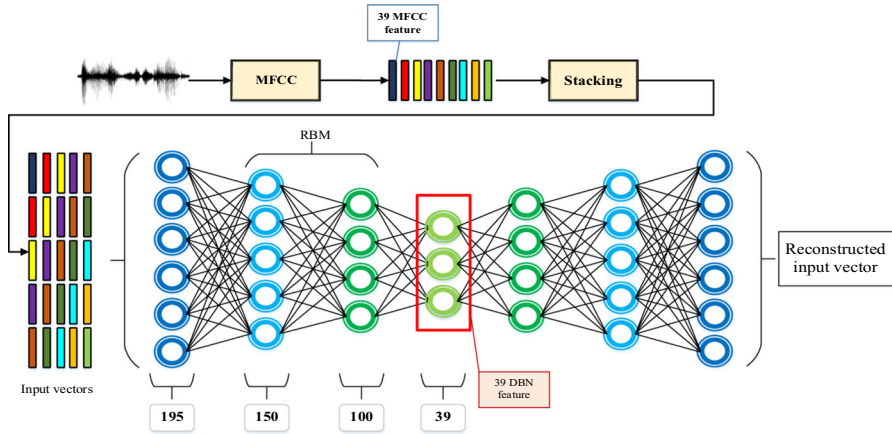


Fig. 4 Architecture of proposed DBN to extract efficient features based on the MFCC vector

$$P(v_i|h) = N(v, \sum_j h_j w_{ij} + a_i, \sigma_i^2) \tag{17}$$

where $N(v, \mu, \Sigma)$ denotes a Gaussian distribution of v with a mean vector μ and a covariance matrix Σ . In the unsupervised pre-training, the data is normalized using the CD with the intention that each coefficient vector has a mean and unit variance equal to zero. Whereas the CD is not exact, several other methods, like probabilistic contrastive divergence (PCD), have been proposed in the RBM [5]. Unlike the CD, which utilizes training data as the initial value for visible units, the PCD method uses the last chain state in the last update step. In other words, the PCD employs successive Gibbs sampling runs to estimate $\langle v_i h_j \rangle_{\text{model}}$.

The proposed DBN is illustrated in Fig. 4. DBN training uses the greedy algorithm [23]. Using this algorithm in a three-layer encoder that consists of three RBMs, initially, a single RBM is trained. After training the first RBM, the second RBM joins it and is trained using the first RBM’s output as the second RBM’s input, and this process continues until the end of the encoder part. After this, the decoder, which is the reverse of the encoder, concatenates to it, and the error backpropagation algorithm is performed for the final correction of the weights.

In the arrangement of the proposed DBN, the number of layers and neurons of each layer is selected so that the middle layer reaches a convergence about speaker information and consequently retrieves the input information in the last layer based upon the middle layer’s converged information.

During three restricted Boltzmann machines, information convergence occurs by reducing the number of neurons in each layer. Therefore, it acts as an encoder and produces low-dimensional features. The second half of this network, which is the reverse of the first part, tries to retrieve the input data based on the low-dimensional features of the middle layer like a decoder. The training continues until the network can accurately reconstruct the information in the output layer. If the network can

properly model the input vector with a low number of middle-layer neurons and then reconstruct the input data, the information of this middle layer can accurately describe the input vector so that it can be used as a low-dimensional feature containing important information of the speech signal. This is precisely the property of an excellent feature extraction algorithm.

4 Experimental Setup

In this section, we explain all the experimental settings. These settings include the datasets, front-end, baseline system, proposed DBN-based system parameters, and evaluation metrics.

4.1 Datasets

The NIST SRE2004 data was used to train the UBM, DBN, and T matrix. This data consisted of 10,743 telephone speech audio files from 480 speakers (181 males and 299 females) [1]. The NIST SRE2008 data was utilized to evaluate the systems. In SRE2008 data, the interview speeches were recorded with several types of microphones in addition to the telephone speech [26]. The utterances of 1270 speakers from all languages in the dataset in both interview and telephone scenarios were used to assess the systems. Training and testing conditions were performed based on predefined Short2 and Short3 conditions, respectively. Speech files contained two-channel telephone conversations of approximately five minutes of target speaker. However, for the interview segments, approximately a three-minute involved target speaker.

4.2 Front-end

A voice activity detector (VAD) in the front-end block separated the speech and silence sections [35]. From an enhanced speech, silence can be detected with higher accuracy. Various methods have been proposed to improve speech signals; among them, the spectrum-based methods stand more attentive [40]. This study used an energy-based VAD called spectral subtraction voice activity detection (SSVAD) to perform speech enhancement and silence removal [6]. SSVAD is specially designed for NIST datasets. In the SRE2008 dataset, in addition to telephone data, interview data, which have a lower signal-to-noise ratio than telephone data, was also included. Therefore, this specialized VAD is considered for this dataset in this work. It has also been shown that using this VAD has been associated with increase in the efficiency of speaker verification systems [35]. In the next step, the first 12 coefficients of MFCC, the energy coefficient, and the first and second derivatives were extracted from speech. This process was performed on 25 ms of speech frame length with 10 ms intervals using the HTK toolbox [57].

This work uses two feature normalization methods: Cepstral Mean and Variance Normalization (CMVN) and Cepstral Mean and Variance Normalization over a sliding

window (WCMVN) that typically spans 301 frames to remove the linear channel effects.

4.3 Baseline System

Baseline systems include the GMM-UBM and i-vector-PLDA speaker verification systems based on the MFCC features. The i-vector-PLDA-based system uses the same UBM trained in the GMM-UBM system. Through LDA, the dimensionality of the vector was reduced to 150, and the PLDA scoring was utilized.

4.4 Proposed DBN-Based System

The optimal parameters for the DBN, such as the input type, the number of layers, and the number of input neurons, were determined based on our computational resources during several experiments. Various features like the time-domain speech signal, the Fourier transform of the signal, and the MFCC were considered as inputs. Consequently, the best speaker verification performance was obtained using the MFCC as the DBN input.

Regarding the arrangement of DBN's input strategy, the best results were achieved when the DBN input contained five consecutive frames of the MFCC. To compare the performance of the proposed DBN-based and MFCC features, the dimension of feature vectors was considered 39. The DBN was trained with half a million data samples (half of the data concerning NIST SRE2004 and NIST SRE2008).

The proposed DBN consists of six RBM blocks with two GRBM layers as the input and output to model speech data. As shown in Fig. 3, the input layer contains 195 neurons to receive five frames of the 39-length MFCC feature. The middle layer consists of 39 neurons to extract the DBN feature. Two layers are arranged between the input and middle layers with 150 and 100 neurons, respectively.

The proposed DBN model was trained in two stages. First, unsupervised learning was performed for 100 epochs using the PCD method. Then, error backpropagation was conducted with a maximum of 200 iterations using the mean squared error (MSE) loss function. The training process utilized a batch size of 100, a learning rate of 0.001, and a penalty of 0.0002.

The DBN was applied to both the GMM-UBM and i-vector-PLDA systems, and the results were investigated under the same conditions with and without the DBN block. In other words, we reported the results of all four conditions to show the impact of DBN on the performance of typical speaker verification systems. The MSR and DeeBNet toolboxes have been employed in the system implementation [32, 45].

4.5 Evaluation Criteria

In this research, the detection error trade-off (DET) curve and detection cost function (DCF) are utilized as the metrics to evaluate the systems. Generally, there are four types of decisions for a test utterance. These errors are defined as follows:

False Positive (FP): Accept fake speaker (incorrectly accept).

False Negative (FN): Reject target speaker (incorrectly reject).

True Positive (TP): Accept target speaker (correctly accept).

True Negative (TN): Reject fake speaker (correctly reject).

According to these definitions, there are two types of errors in speaker verification systems. These error coefficients are defined as follows:

$$\text{PFR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (18)$$

$$\text{FNR} = \frac{\text{FN}}{\text{FN} + \text{TP}} \quad (19)$$

$$\text{EER} = \frac{\text{FNR} + \text{FPR}}{2} \quad \text{if } \text{FNR} = \text{FPR} \quad (20)$$

The Equal Error Rate (EER) is where the FPR and FNR errors are equal, obtained by changing the decision threshold. DET is a graphical scheme that plots FNR versus FPR. The speaker verification systems obtained a matching coefficient between the trained acoustic models and the test utterances. This score is a variable, indicating the similarity between the trained speaker and the test speaker. A high score indicates greater similarity. The system requires a threshold value to make a decision. If this decision threshold value is low, the FA error increases; otherwise, it increases the FR error. In the DET plot, the higher the system performance, the closer the curve to zero. The detection cost function (DCF) is defined as a weighted sum of miss and false alarm errors, where the cost function is minimum.

$$\begin{aligned} \text{DCF} = & C_{\text{miss}} \times P_{\text{Miss}|\text{Target}} \times P_{\text{Target}} \\ & + C_{\text{FalseAlarm}} \times P_{\text{FalseAlarm}|\text{NonTarget}} \times (1 - P_{\text{Target}}) \end{aligned} \quad (21)$$

The DCF is calculated via the parameter value $C_{\text{Miss}} = 10$, $C_{\text{FalseAlarm}} = 1$, $P_{\text{Target}} = 0.01$ for the dataset NIST 2008 [26].

5 Evaluations and Results

This section presents the results of the baseline and the proposed systems for short utterance text-independent speaker verification. Various parameters were examined to design these systems. This variety includes the presence and absence of the SSVAD, the determination of the best selection for the number of GMM components, and the feature normalization method. The results are listed in Table 1. All the systems were tested on 1270 speakers (telephone and interview) of the NIST2008 dataset. Initially, a system with 512 Gaussian mixtures was designed without considering any of the proposed methods and even without applying the SSVAD. This experiment extracted the MFCC features without applying the SSVAD and used them for GMM-UBM training and testing. In this condition, the EER was equal to 33.2%. The SSVAD was employed to remove silence and improve the result. Under these conditions, firstly, the

Table 1 The results of the GMM-UBM systems in different steps of the design

System configuration	EER (%)	DCF08
512GMM + MFCC	33.20	0.1000
1024GMM + MFCC + SSVAD	25.19	0.0980
512GMM + MFCC + SSVAD	25.15	0.0976
512GMM + DBN	15.50	0.0652
512GMM + CMVN + DBN	14.34	0.0604
512GMM + WCMVN + DBN	14.30	0.0600

signals were denoised, silent parts were removed, and then MFCC feature extraction was followed. The GMM with 512 and 1024 mixtures was evaluated to determine the optimal number of the mixtures. Considering the SSVAD, the EER value for the 512 and 1024 mixtures was reported as 25.15% and 25.19%, respectively. As can be seen, the SSVAD improves the system performance by about 8%. It can also be found that 512 components for the GMM produce better results and have less computational cost.

Adding the proposed DBN feature reduced EER to 15.50% and improved the system performance by about 10%. It can be seen that using the proposed DBN can improve the final result in a typical GMM-UBM system. Because of the benefits of feature normalization methods in speaker verification, the CMVN and WCMVN were employed. The MFCC features were normalized and utilized to train the DBN.

The EER benchmark reached 14.34% and 14.30% in the case of the CMVN and the WCMVN, respectively. In this comparison, WCMVN showed the lowest EER. To illustrate the impact of using the DBN, Fig. 5 shows the DET curves for the two modes before and after using DBN.

The second scenario discusses the performance of the i-vector-PLDA-based system with 512 GMM components with 256 i-vector lengths. The system was designed in

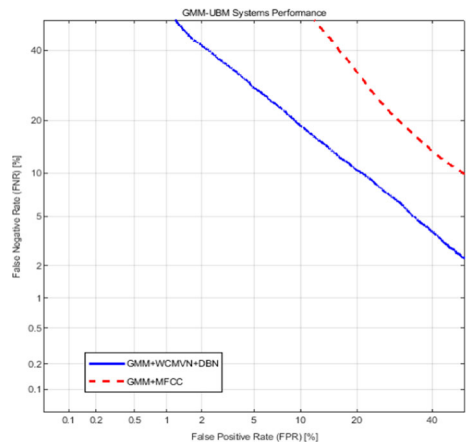
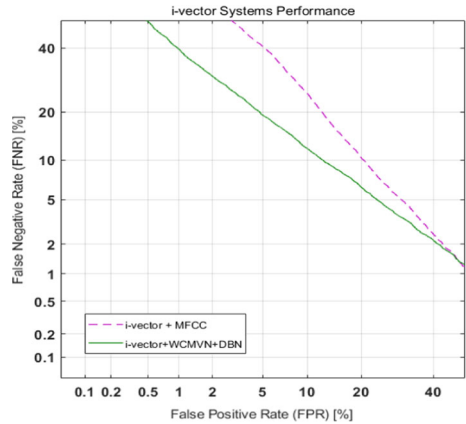
Fig. 5 The DET curve to compare the two GMM-UBM based systems

Table 2 The results of the i-vector system in two scenarios with and without DBN

System configuration	EER (%)	DCF08
i-vector + MFCC	15.24	0.0732
i-vector + WCMVN + DBN	10.97	0.0484

Fig. 6 The DET curve of the two systems based on i-vector/PLDA



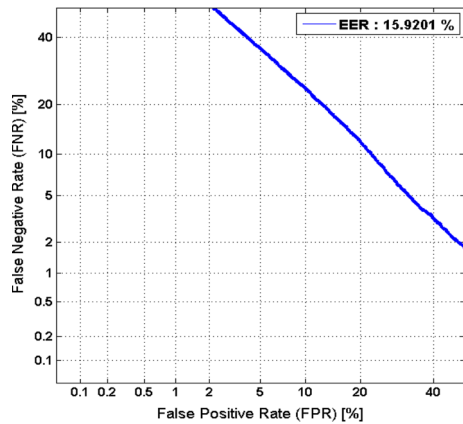
two modes, with and without the DBN. The results of these systems are shown in Table 2. In addition, the DET curve is indicated in Fig. 6.

The i-vector-PLDA system achieved an EER value equal to 15.24% when the MFCC feature was utilized. Adding the WCMVN and DBN methods results in an EER value of 10.97%. Applying the DBN and WCMVN methods showed that the i-vector-based system improved the performance by up to 4.27% in the EER metric. We can see the proposed generative model's impact on popular speaker verification systems.

The evaluation of the systems on this scale has a high computational cost and requires an extended processing time. In many applications, the processing time is a priority, even if system performance declines. Moreover, people are less inclined to provide long training speeches in real-world applications. Feature extraction can have an essential role in solving the computational cost problem. Therefore, besides using the DBN as a feature extractor, this scenario attempted to reduce the data dimension to decrease the processing time.

The previous experiments utilized five consecutive MFCC frames for training the DBN with one frame interval at each step so that the DBN tries to model the middle frame. This experiment used five consecutive speech frames with five frame intervals per step. Under these conditions, the DBN tries to model five frames per step, and there would be no overlap between the MFCC of each step. Using this method, the DBN reduced the volume of data by one-fifth. During the evaluation phase, each audio file of the dataset remained a speech for about 3 min after applying the SSVAD algorithm. Based on proposed DBN dimension-reduction method, there is a signal with about 30 s duration. The result of this system is shown in Fig. 7. Under these circumstances, the computational time has been significantly reduced to approximately one-twelfth

Fig. 7 The DET curves while the DBN network extracts features and reduces data dimension



compared to the previous system, which utilized i-vector + WCMVN + DBN. This means that the new speaker verification system, incorporating the DBN-based feature extraction and dimension-reduction strategy, takes around 16 min to analyze and make decisions on all the test files for 1270 speakers. This computations were performed using a single CPU (i7 ten-generation) and 32GB RAM as the available computational resources. The EER benchmark reached 15.92%, which is 4.95% higher than the case in which dimension reduction was not applied. Reducing processing time is crucial for various tasks. The system can significantly decrease the processing load by employing dimension-reduction strategies. This enables the implementation of the system in real-world applications with limited computational resources.

6 Conclusion and Future Works

The present research showed the highly improved performance of the GMM-UBM-based and i-vector-PLDA-based speaker verification systems in the text-independent mode with the proposed DBN features. By comparing the results of baseline MFCC and proposed DBN-based systems, it was found that the proposed generative network improves performance. In another scenario, DBN was used to feature dimension reduction and decrease the computational time. The proposed feature dimension reduction reduced the length of the utterances and made light systems suitable for some online scenarios and devices with low computational resources, such as mobile phones. Hopefully, future studies can improve performance by implementing the proposed feature extraction method into a system with new acoustic modeling, datasets, and feature normalization methods. Moreover, utilizing a convolutional neural network in autoencoder architecture can be a future trend in speaker-specific feature extraction. The source code of this paper is available from [15].

Author Contributions Aref Farhadipour wrote the manuscript's text, figures, tables, and simulations. Hadi Veisi was the supervisor and reviewer in all stages.

Funding Open access funding provided by University of Zurich. The authors did not receive support from any organization for the submitted work.

Availability of Data and Materials The source code of this paper is available from [15]. The NIST 2004 and 2008 datasets are available on the NIST SRE official website upon request.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Ethical Approval This paper reflects the authors' own research and analysis truthfully and completely and is not currently being considered for publication elsewhere.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. M.P. Alvin, A. Martin, NIST speaker recognition evaluation chronicles. In: The Speaker and Language Recognition Workshop (ODYSSEY, 2004)
2. L Alzubaidi J Bai A Al-Sabaawi J Santamaría A Albahri BSN Al-dabbagh MA Fadhel M Manoufali J Zhang AH Al-Timemy 2023 A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications J. Big Data 10 46 127
3. Z Bai XL Zhang 2021 Speaker recognition based on deep learning: an overview Neural Netw. 140 65 99
4. A. Banerjee, A. Dubey, A. Menon, S. Nanda, G.C. Nandi, Speaker recognition using deep belief networks. [arXiv:1805.08865](https://arxiv.org/abs/1805.08865) (2018)
5. IBisio F Lavagetto C Garibotto A Sciarrone 2017 Speaker recognition exploiting D2D communications paradigm: performance evaluation of multiple observations approaches Mob. Netw. Appl. 22 1045 1057
6. S Boll 1979 Suppression of acoustic noise in speech using spectral subtraction IEEE Trans. Acoust. Speech Signal Process. 27 113 120
7. T. Chen, E. Khoury, Speaker embedding conversion for backward and cross-channel compatibility. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7072–7076 (2022)
8. A. Chowdhury, A. Cozzo, A. Ross, Domain adaptation for speaker recognition in singing and spoken voice. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7192–7196 (2022)
9. S Cumani O Pichot P Laface 2014 On the use of i-vector posterior distributions in probabilistic linear discriminant analysis IEEE Trans. Audio Speech Lang. Process. 22 846 857
10. RK Das SM Prasanna 2018 Speaker verification from short utterance perspective: a review IETE Tech. Rev. 35 599 617
11. N Dehak PJ Kenny R Dehak P Dumouchel P Ouellet 2010 Front-end factor analysis for speaker verification IEEE Trans. Audio Speech Lang. Process. 19 788 798
12. B. Desplanques, J. Thienpondt, K. Demuynck, Ecapa-tdnn: emphasized channel attention, propagation and aggregation in tdnn based speaker verification. [arXiv:2005.07143](https://arxiv.org/abs/2005.07143) (2020)
13. M Dua C Jain S Kumar 2022 LSTM and CNN based ensemble approach for spoof detection task in automatic speaker verification systems J. Ambient Intell. Hum. Comput. 13 1985 2000

14. SA El-Moneim M Nassar MI Dessouky NA Ismail AS El-Fishawy FEA El-Samie 2022 Cancellable template generation for speaker recognition based on spectrogram patch selection and deep convolutional neural networks *Int. J. Speech Tech.* 25 689 696
15. A. Farhadipour, ivector and GMMUBM based speaker verification MATLAB code. https://github.com/areffarhadi/iVector_GMMUBM_Speaker_Verification (2024)
16. A Farhadipour H Veisi M Asgari MA Keyvanrad 2018 Dysarthric speaker identification with different degrees of dysarthria severity using deep belief networks *Etri J.* 40 643 652
17. JL Gauvain CH Lee 1994 Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains *IEEE Trans. Speech Audio Process.* 2 291 298
18. O. Ghahabi, J. Hernando, Deep belief networks for i-vector based speaker recognition, In: *The IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1700–1704 (2014)
19. J Guo N Xu K Qian Y Shi K Xu Y Wu A Alwan 2018 Deep neural network based i-vector mapping for speaker verification using short utterances *Speech Commun.* 105 92 102
20. G Hinton L Deng D Yu GE Dahl AR Mohamed N Jaitly A Senior V Vanhoucke P Nguyen TN Sainath 2012 Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups *IEEE Signal Process. Mag.* 29 82 97
21. GE Hinton 2002 Training products of experts by minimizing contrastive divergence *Neural Comput.* 14 1771 1800
22. GE Hinton 2009 Deep belief networks *Scholarpedia* 4 5947
23. GE Hinton S Osindero YW Teh 2006 A fast learning algorithm for deep belief nets *Neural Comput.* 18 1527 1554
24. GE Hinton RR Salakhutdinov 2006 Reducing the dimensionality of data with neural networks *Science* 313 504 507
25. J.W. Jung, H. Tak, H.J. Shim, H.S. Heo, B.J. Lee, S.W. Chung, H.G. Kang, H.J. Yu, N. Evans, T. Kinnunen, SASV challenge 2022: a spoofing aware speaker verification challenge evaluation plan. [arXiv:2201.10283](https://arxiv.org/abs/2201.10283) (2022)
26. S.S. Kajarekar, N. Scheffer, M. Graciarena, E. Shriberg, A. Stolcke, L. Ferrer, T. Bocklet, The SRI NIST 2008 speaker recognition evaluation system. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4205–4208 (2009)
27. A Kanagasundaram D Dean S Sridharan J Gonzalez-Dominguez J Gonzalez-Rodriguez D Ramos 2014 Improving short utterance i-vector speaker verification using utterance variance modelling and compensation techniques *Speech Commun.* 59 69 82
28. A. Kanagasundaram, S. Sridharan, S. Ganapathy, P. Singh, C. Fookes, A study of x-vector based speaker recognition on short utterances. In: *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2943–2947 (2019)
29. V Karthikeyan 2022 Modified layer deep convolution neural network for text-independent speaker recognition *J. Exp. Theo. Artif. Intell.* 36 1 13
30. P Kenny G Boulianne P Ouellet P Dumouchel 2007 Joint factor analysis versus eigenchannels in speaker recognition *IEEE Trans. Audio Speech Lang. Process.* 15 1435 1447
31. P. Kenny, T. Stafylakis, P. Ouellet, M.J. Alam, P. Dumouchel, PLDA for speaker verification with utterances of arbitrary duration. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7649–7653 (2013)
32. M.A. Keyvanrad, M.M. Homayounpour, A brief survey on deep belief networks and introducing a new object oriented toolbox (DeeBNet). [arXiv:1408.3264](https://arxiv.org/abs/1408.3264) (2014)
33. WB Kheder D Matrouf M Ajili JF Bonastre 2018 A unified joint model to deal with nuisance variabilities in the i-vector space *IEEE Trans. Audio Speech Lang. Process.* 26 633 645
34. L. Li, D. Wang, W. Du, D. Wang, CP map: a novel evaluation toolkit for speaker verification. [arXiv:2203.02942](https://arxiv.org/abs/2203.02942) (2022)
35. MW Mak HB Yu 2014 A study of voice activity detection techniques for NIST speaker recognition evaluations *Comput. Speech Lang.* 28 295 313
36. M McLaren D Leeuwen Van 2011 Source-normalized LDA for robust speaker recognition using i-vectors from multiple speech sources *IEEE Trans. Audio Speech Lang. Process.* 20 755 766
37. TK Moon 1996 The expectation-maximization algorithm *IEEE Signal Process. Mag.* 13 47 60
38. AB Nassif I Shahin A Elnagar D Velayudhan A Alhudhaif K Polat 2022 Emotional speaker identification using a novel capsule nets model *Expert Syst. Appl.* 193 116469

39. D. Nongrum, F. Pyrtuh, A comparative study on effect of temporal phase for speaker verification. In: *Proceedings of International Conference on Frontiers in Computing and Systems (COMSYS)*, pp. 571–578 (2021)
40. PG Patil TH Jaware SP Patil RD Badgujar F Albu I Mahariq B Al-Sheikh C Nayak 2022 Marathi speech intelligibility enhancement using I-AMS based neuro-fuzzy classifier approach for hearing aid users *IEEE Access* 10 123028 123042
41. A Poddar M Sahidullah G Saha 2018 Speaker verification with short utterances: a review of challenges, trends and opportunities *IET Biom.* 7 91 101
42. A Poddar M Sahidullah G Saha 2019 Quality measures for speaker verification with short utterances *Digit. Signal Process.* 88 66 79
43. DA Reynolds TF Quatieri RB Dunn 2000 Speaker verification using adapted Gaussian mixture models *Digit. Signal Process.* 10 19 41
44. F Richardson D Reynolds N Dehak 2015 Deep neural network approaches to speaker and language recognition *IEEE Signal Process. Lett.* 22 1671 1675
45. SO Sadjadi M Slaney L Heck 2013 MSR identity toolbox v1.0: a MATLAB toolbox for speaker-recognition research *Speech Lang. Process. Techn. Comm. Newsl.* 1 1 32
46. S Saleem F Subhan N Naseer A Bais A Imtiaz 2020 Forensic speaker recognition: a new method based on extracting accent and language information from short utterances *Forens. Sci. Int. Digit. Investig.* 34 300982
47. L Sun T Gu K Xie J Chen 2019 Text-independent speaker identification based on deep Gaussian correlation supervector *Int. J. Speech Tech.* 22 449 457
48. D. Sztahó, G. Szaszák, A. Beke, Deep learning methods in speaker recognition: a review. [arXiv:1911.06615](https://arxiv.org/abs/1911.06615) (2019)
49. H. Tak, M. Todisco, X. Wang, J.W. Jung, J. Yamagishi, N. Evans, Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation. [arXiv:2202.12233](https://arxiv.org/abs/2202.12233) (2022)
50. M. Takamizawa, S. Tsuge, Y. Horiuchi, S. Kuroiwa, Same speaker identification with deep learning and application to text-dependent speaker verification. In: *Human Centred Intelligent Systems Conference*, pp. 149–158 (2022)
51. Y. Tang, G. Ding, J. Huang, X. He, B. Zhou, Deep speaker embedding learning with multi-level pooling for text-independent speaker verification. In: *IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 6116–6120 (2019)
52. F. Tong, M. Zhao, J. Zhou, H. Lu, Z. Li, L. Li, Q. Hong, ASV-subtools: open source toolkit for automatic speaker verification. In: *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6184–6188 (2021)
53. E. Variani, X. Lei, E. McDermott, I. L. Moreno, J. Gonzalez-Dominguez, Deep neural networks for small footprint text-dependent speaker verification. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4052–4056 (2014)
54. S. Wang, J. Rohdin, L. Burget, O. Plhot, Y. Qian, K. Yu, J. Cernocký, On the usage of phonetic information for text-independent speaker embedding extraction. In: *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1148–1152 (2019)
55. X. Wang, L. Li, D. Wang, VAE-based domain adaptation for speaker verification. In: *The Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pp. 535–539 (2019)
56. Z. Wu, S. Wang, Y. Qian, K. Yu, Data augmentation using variational autoencoder for embedding based speaker verification. In: *Proceedings of the Annual Conference of the International Speech Communication Association, (INTERSPEECH)*, pp. 1163–1167 (2019)
57. S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, *The HTK book*. Cambridge University Engineering Department (2002)
58. Y.Q. Yu, W.J. Li, Densely connected time delay neural network for speaker verification. In: *Proceedings of the Annual Conference of the International Speech Communication Association, (INTERSPEECH)*, pp. 921–925 (2020)
59. Y Zhao R Togneri V Sreeram 2022 Multi-task learning-based spoofing-robust automatic speaker verification system *Circuits Syst. Signal Process.* 41 4068 4089