Speaker Recognition by means of Restricted Boltzmann Machine Adaptation

Pooyan Safari, Omid Ghahabi, Javier Hernando
pooyan.safari@tsc.upc.edu, {omid.ghahabi,javier.hernando}@upc.edu
TALP Research Center, Department of Signal Theory and Communications
Universitat Politècnica de Catalunya - BarcelonaTech, Spain

***Abstract*—Restricted Boltzmann Machines (RBMs) have shown success in speaker recognition. In this paper, RBMs are investigated in a framework comprising a universal model training and model adaptation. Taking advantage of RBM unsupervised learning algorithm, a global model is trained based on all available background data. This general speaker-independent model, referred to as URBM, is further adapted to the data of a specific speaker to build speaker-dependent model. In order to show its effectiveness, we have applied this framework to two different tasks. It has been used to discriminatively model target and impostor spectral features for classification. It has been also utilized to produce a vector-based representation for speakers. This vector-based representation, similar to i-vector, can be further used for speaker recognition using either cosine scoring or Probabilistic Linear Discriminant Analysis (PLDA). The evaluation is performed on the core test condition of the NIST SRE 2006 database.**

I. Introduction

Gaussian Mixture Models (GMMs) are the core of many state-of-the-art speaker recognition systems. They are used in the conventional GMM-UBM approach in an adaptation process to model speakers. This adaptation is carried out by means of Maximum a Posteriori (MAP) estimation. A high-dimensional vector, called supervector, is formed by concatenating the mean vectors obtained from the MAP-adapted GMMs. The dimension of these supervectors are further reduced using an effective Factor Analysis (FA) technique renowned as i-vector [1]. These i-vectors can be employed for classification in speaker recognition applications using cosine distance similarity or Probabilistic Linear Discriminant Analysis (PLDA) [1]–[3].

Restricted Boltzmann Machines (RBMs) are generative models able to efficiently learn via unsupervised learning algorithms. They have recently shown success in applications such as audio and speech processing (e.g., in [4]–[6]). In speaker recognition, they were used to extract features [7], and speaker factors [8], and to classify i-vectors [9], [10]. They have been employed in an adaptation process [11]–[14], to further discriminatively model target and impostor speakers. RBMs have been recently used in DBNs as a pre-training stage to extract Baum-Welch statistics for i-vector and supervector extraction [15], [16]. RBMs were used in [17] prior to PLDA, as a transformation stage of i-vectors, to build a more suitable discriminative representation for the supervised classifier. It is also worth noting that recently different methods have been proposed to incorporate Deep Neural Networks (DNNs) into the context of speaker recognition. In [18], [19], DNNs were used to extract an enriched vector representation of speakers for text-dependent speaker verification. In [20], DNNs were employed to extract a more discriminative vector from i-vector. They have been used in [21] to collect sufficient statistics for i-vector extraction. There were also few attempts addressing methods to produce alternative vector-based speaker representation using RBMs [22]–[24].

In this paper, a framework is investigated including two stages, namely Universal RBM (URBM) training and model adaptation. This framework is applied to two different tasks in order to show the efficiency of the method. In the first application, a speaker recognition system is designed to set up discriminative target speaker models, using speaker spectral features. In the second task, speaker spectral features are mapped into a single fixed-dimensional vector conveying speaker-specific information. This new vector-based representation will be referred to as RBM-vector, and can be further used in speaker recognition by either cosine scoring or Probabilistic Linear Discriminant Analysis (PLDA). It will be shown that the proposed framework outperforms the conventional approaches in each application.

II. Universal Model

As it is illustrated in Fig. 1, the first step of the proposed framework is to train a universal model based on all available background data, which conveys the speaker-independent information. Taking advantage of RBM unsupervised learning algorithm, a global model is trained, referred to as Universal RBM (URBM). This universal model is built by training a single RBM given the speaker spectral features, extracted from all background utterances. The binary hidden units are chosen for the RBM. However, due to the fact that the features are real-valued data, we use Gaussian real-valued units for observed variables. RBMs can be trained using an approximated version of Contrastive Divergence (CD) algorithm called CD-1 [25]. The CD-1 algorithm for Gaussian-Bernoulli RBMs works under the assumption that the inputs have zero mean and unit variance [26]. Therefore, a cepstral mean-variance normalization (CMVN) is applied to the features of each utterance prior to RBM training. URBM represents the general, speaker-independent model. It is assumed that URBM is able to learn both speaker and session variabilities from background data. It should be built using whole available background samples (feature vectors) in order to cover a wide range of speaker and channel variabilities. However, due to resource limitations we randomly select as many background feature vectors as possible for training.

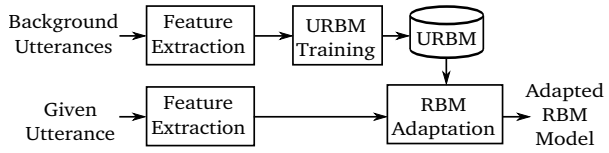

Figure 1: Block diagram of the proposed framework. Feature extraction also includes a speaker dependent cepstral mean-variance normalization. Adapted RBM models can be further used for different purposes.

III. Model Adaptation

In order to build a speaker-specific model for each speaker, it is proposed to incorporate speaker-dependent information into the obtained universal model (URBM). This is carried out by means of an adaptation stage. For each speaker, the speaker adaptation is performed by training an RBM model with a few number of iterations using data samples of the corresponding speaker. The parameters of this RBM model, such as weights and biases, are initialized by the ones obtained from the URBM. In other words, the URBM is adapted to the data of each speaker. The idea of this kind of adaptation has also shown success in [11]–[14], [24] to initialize the parameters of DNNs for classification purposes. This speaker adaptation, modifies the weights of the universal model. It should be noted that in comparison with URBM training, fewer number of epochs is used for the adaptation procedure. This is important in order to avoid overfitting. This also makes the training time much less than what is needed for training a speaker-specific model without adaptation.

IV. Speaker Feature Classification

In this section the above mentioned technique has been incorporated into a classification task for speaker recognition.The block diagram of a speaker recognition system is shown in Fig. 2. This is a modified version of the system which has been proposed in [14]. The spectral features are extracted and subject to a speaker-dependent mean-variance normalization. Background feature vectors are used to build the Universal RBM (URBM) model. Before training the discriminative target model, a random sample selection is applied to the background data. The same impostor samples are selected for all target speakers in order to perform discriminative training using target and impostor labels. The number of selected impostor samples is almost equal to the average number of target samples. On the other hand, the number of target samples varies from one speaker to another. We fix the number of minibatches for all target speakers instead of using fixed minibatch size. In this way, the number of times that the parameters of each network is updated in each iteration (epoch) will remain constant for all target speakers.

As it was mentioned before, the aim in the URBM/RBM-adaptation framework is to capture the speaker-independent information from all available background data by training an RBM, and then to adapt the background model to few available data of each target speaker. This URBM can also tackle the imbalance between the two classes of impostor and target speaker samples by incorporating the information lies in the huge amount of impostor data in a single universal model. The URBM should be built on the whole available impostor samples. However, due to resource limitations we select randomly as many impostor samples as possible.

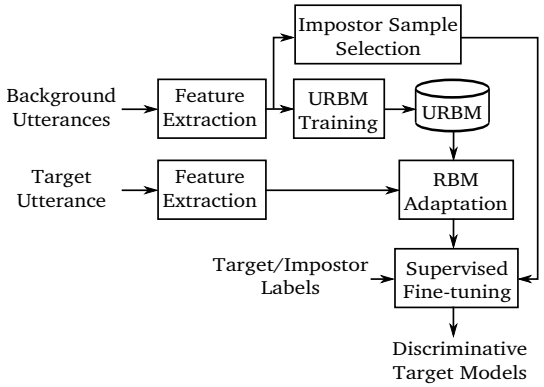

Figure 2: Block diagram of the system used in scenario one. Feature extraction also includes a speaker dependent cepstral mean-variance normalization.

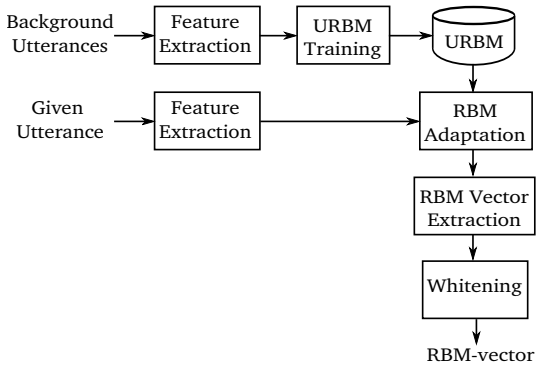

Figure 3: Block diagram showing different stages of RBM-vector extraction process. Feature extraction also includes a speaker dependent cepstral mean-variance normalization.

V. Speaker Vector Representation

The proposed framework is also used in order to produce an alternative vector-based representation for speakers, which will be referred to as RBM-vectors. Figure 3 shows the block diagram of the RBM-vector extraction process. Employing speech spectral features, an RBM model is trained based on background data. It is then adapted to the data of each speaker in order to build a model per speaker. This speaker adaptation, modifies the weights of the universal model. These models are adopted to form RBM-vactors. These vectors can be further used for speaker verification.

Once the adaptation step is completed, an RBM model is assigned to each speaker. The parameters of these models such as hidden-visible connection weights $\mathbf{W}$ and biases can be employed to build the speaker's vector. The weights carry speaker-specific information which are distinct enough one from another, to be used for speaker recognition. The rows of the weight matrix along with the bias vectors are concatenated to form a high-dimensional RBM supervector. The obtained vectors will be subject to a mean-normalization prior to PCA whitening with dimension reduction. PCA is trained using background speaker vectors and then applied to all the background, target, and test vectors. Whitening transformation rotates the original data to the principal component space in which the rotated data components are less correlated,

$$\mathbf{\Lambda}_{L\times M} = \left(\mathbf{S}_{1:L\times 1:L} + \varepsilon\right)^{-1/2} \mathbf{U}_{1:L\times M} \tag{1}$$

where $\mathbf{\Lambda}$ is the transformation matrix which is multiplied by

Table I: Results obtained on the core test condition of NIST SRE 2006 evaluation. ADBN is referred to the proposed Adapted DBN technique.

Classifier	EER (%)
MLP	18.12
DBN	17.19
ADBN	**16.67**

the original data for whitening and dimension reduction, **U** is the matrix of eigenvectors, **S** is the diagonal matrix of the corresponding eigenvalues, M and L are the values for the dimension of original and shortened vectors, respectively. A small constant of ε is added, as a regularization factor, to avoid large values in practice. The values for L and ε must be set experimentally to optimize the results.

The output of the whitening stage is called the RBM-vector and similar to i-vector can be used for speaker verification using cosine similarity or PLDA. In the next section, it will be shown that using weights to build the speaker-specific vectors is able to outperform the conventional i-vector approach.

VI. Experimental Results

A. Database and setup

Frequency Filtering (FF) [27] features have been used for the experiments. FF features, like MFCCs, are a decorrelated version of FBEs [27]. It has been shown that FF features achieve equal or better performance than MFCCs [27]. They are extracted every 10 ms with a 30 ms Hamming window. The size of static FF features is 16. Before feature extraction, speech signals are subject to an energy-based silence removal process. The whole core test condition of the NIST 2006 SRE evaluation [28] is considered in all the experiments. It comprises 816 target speakers, with $51,068$ trials. Each signal consists of about two minutes of speech. Performance is evaluated using the Equal Error Rate (EER) calculated using $C_M = 10$, $C_{FA} = 1$, and $P_T = 0.01$.

We use 5 neighbouring frames (2-1-2) of the features in order to compose 80-dimensional feature inputs for all networks. The RBMs used for the speaker feature classification comprise 128 hidden units. However, for the speaker vector representation task, we have employed RBMs with 400 hidden units. The URBM is trained by a learning rate of 0.0001 with 200 epochs with minibatch size of 100. The URBM should be trained based on all available background data which is here about 60 million feature vectors. However, due to the resource limitations we have done a random sample selection prior to training and reduced the number of feature vectors to 4 and 8 million, for the first and second scenarios, respectively. Adaptation process is carried out by 5 epochs of CD-1 algorithm and a learning rate of 0.001 for the first, and 0.005 for the second scenario. Momentum and weight decay have been used in all the networks.

In the first scenario, the conventional Multilayer Perceptron (MLP), has been considered as our baseline system. It is trained with a learning rate of 0.05 with 400 epochs. For the fine-tuning of RBM target models we have employed a learning rate of 0.09 with 150 epochs and fixed the number of minibatches to 200.

In the second scenario for the i-vector baseline system, the gender-independent UBM is represented as a diagonal covariance, 512-component GMM. ALIZE open source software [29] is used to extract 400-dimensional i-vectors. The development data includes 6125 speech files collected from NIST 2004 and 2005 SRE corpora. It is worth noting that in the case of NIST 2005 only the speech files of those speakers which do not appear in NIST 2006 database are used. The PLDA for the i-vector/PLDA baseline is trained with 15 iterations and the number of eigenvoices is empirically set to 250. It should be mentioned that both i-vectors and RBM-vectors are length-normalized prior to training PLDA. The optimum weights for the score-level fusion has been set manually. For cosine scoring, these weights were set to 0.35 and 0.65 for i-vector and RBM-vector, respectively. In the case of PLDA, they were set to 0.65 and 0.35 for i-vector and RBM-vector, respectively.

Table II: Comparison of the i-vector with different RBM-vectors in terms of EER% and vector dimension using cosine distance. The fusion is applied on score level.

Technique	EER (%)
i-vector (400)	7.01
RBM-vector (400)	7.26
RBM-vector (600)	6.77
RBM-vector (800)	6.58
RBM-vector (2000)	5.98
Fusion i-vector (400) & RBM-vector (2000)	**5.30**

Table III: Comparison of the performance of PLDA with i-vector, and RBM-vectors of different dimensions, in terms of EER%. The fusion is applied on the score level.

Technique	EER (%)
i-vector (400)	4.90
RBM-vector (400)	5.55
RBM-vector (600)	5.15
RBM-vector (800)	5.42
i-vector (400)+RBM-vector (600) Fusion	**4.21**

B. Results

RBM-vector has been evaluated using cosine similarity. The results are shown in Table II. The performance of RBM-vector of size 400 is comparable to the i-vector of equal length. The last row in the table shows the score-level fusion of the i-vector technique and RBM-vector of size 400, which is more than 24% relative improvement compared to using only i-vector. This is important particularly when no data label is available to perform supervised compensation techniques such as PLDA.

PLDA is also applied to RBM-vectors and the results have been reported in Table III. The PLDA is trained with 15 iterations and the number of eigenvoices are empirically set to 250, 350, 400, for RBM-vectors of sizes 400, 600, 800, respectively. The RBM-vectors are subject to length normalization prior to PLDA training. Using i-vector/PLDA shows an improvement of about 30% compared to i-vector/cosine framework. Comparing the results obtained by RBM-vector/PLDA framework with the ones from RBM-vector/cosine shows a relative improvement of 24%, 24%, and 18% for RBM-vectors of dimensions 400, 600, and 800, respectively. This reveals that PLDA as a compensation technique, is more suitable for i-vectors than RBM-vectors. This proposes a potential research direction to find more suitable compensation techniques for RBM-vectors.

VII. Conclusions

We have studied a framework which is efficient for speaker recognition system. It is composed of two different stages. In the first step, a universal model is built based on background data, using Restricted Boltzmann Machines (RBMs). This global model, which is referred to as Universal RBM (URBM), carries speaker-independent model. In the next step, URBM is adapted to the data of a given utterance to build the speaker-dependent model. We have used this technique in two different tasks in order to show its efficiency. It has been used for speaker feature classification. The evaluation on the core test condition of the NIST SRE 2006 database shows that it outperforms the conventional MLP by more than 8% relative improvement. The framework has been also utilized for an alternative vector-based representations of speakers. These vectors can be further used in speaker verification by means of cosine scoring or PLDA. The preliminary results on the core test condition of the NIST SRE 2006 database show that this new vector representation outperforms the conventional i-vector using cosine similarity by 15% relative improvement. The fusion with i-vector using cosine can improve more than 24%. As expected, using PLDA instead of cosine similarity, improves the performance of RBM-vectors by 24% relative improvement in terms of EER. Finally, when fusing the RBM-vector/PLDA scores with the ones obtained by i-vector/PLDA a further improvement of 14% is attained compared to using only i-vector/PLDA.

References

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[2] N. Dehak, R. Dehak, J. Glass, D. Reynolds, and P. Kenny, "Cosine Similarity Scoring without Score Normalization Techniques," in *Proc. Odyssey Speaker and Language Recognition Workshop*, Brno, Czech Republic, June 2010.

[3] P. Kenny, "Bayesian Speaker Verification with Heavy-Tailed Priors," in *Keynote presentation, Odyssey Speaker and Language Recognition Workshop*, Brno, Czech Republic, June 2010.

[4] A.-R. Mohamed and G. Hinton, "Phone Recognition using Restricted Boltzmann Machines," in *Proc. ICASSP*, 2010.

[5] N. Jaitly and G. Hinton, "Learning a Better Representation of Speech Soundwaves using Restricted Boltzmann Machines," in *Proc. ICASSP*, 2011.

[6] Z.-H. Ling, L. Deng, and D. Yu, "Modeling Spectral Envelopes using Restricted Boltzmann Machines and Deep Belief Networks for Statistical Parametric Speech Synthesis," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, no. 10, pp. 2129–2139, 2013.

[7] H. Lee, P. Pham, and A. Ng, "Unsupervised Feature Learning for Audio Classification using Convolutional Deep Belief Networks," in *Advances in neural information processing systems*, 2009.

[8] T. Stafylakis, P. Kenny, M. Senoussaoui, and P. Dumouchel, "PLDA using Gaussian Restricted Boltzmann Machines with Application to Speaker Verification," in *Proc. Interspeech*, Portland, USA, September 2012.

[9] T. Stafylakis and P. Kenny, "Preliminary Investigation of Boltzmann Machine Classifiers for Speaker Recognition," in *Proc. Odyssey Speaker and Language Recognition Workshop*, Singapore, June 2012.

[10] M. Senoussaoui, N. Dehak, P. Kenny, and R. Dehak, "First Attempt of Boltzmann Machines for Speaker Verification," in *Proc. Odyssey Speaker and Language Recognition Workshop*, Singapore, June 2012.

[11] O. Ghahabi and J. Hernando, "Deep Belief Networks for i-Vector Based Speaker Recognition," in *Proc. ICASSP*, Florence, Italy, May 2014.

[12] O. Ghahabi and J. Hernando, "i-Vector Modeling with Deep Belief Networks for Multi-Session Speaker Recognition," in *Proc. Odyssey Speaker and Language Recognition Workshop*, Joensuu, Finland, June 2014.

[13] O. Ghahabi and J. Hernando, "Global Impostor Selection for DBNs in Multi-Session i-Vector Speaker Recognition," in *Advances in Speech and Language Technologies for Iberian Languages*. Springer International Publishing, 2014, pp. 89–98.

[14] P. Safari, O. Ghahabi, and J. Hernando, "Feature Classification by means of Deep Belief Networks for Speaker Recognition," in *Proc. EUSIPCO*, Nice, France, August 2015.

[15] 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 Speaker and Language Recognition Workshop*, Joensuu, Finland, June 2014.

[16] W. Campbell, "using Deep Belief Networks for Vector-Based Speaker Recognition," in *Proc. Interspeech*, Singapore, May 2014.

[17] S. Novoselov, T. Pekhovsky, K. Simonchik, and A. Shulipa, "RBM-PLDA Subsystem for the NIST i-Vector Challenge," in *Proc. Interspeech*, Singapore, May 2014.

[18] E. Variani, X. Lei, E. McDermott, I. Lopez Moreno, and J. Gonzalez-Dominguez, "Deep Neural Networks for Small Footprint Text-Dependent Speaker Verification," in *Proc. ICASSP*, Florence, Italy, May 2014.

[19] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep Feature for Text-Dependent Speaker Verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[20] Y. Isik, H. Erdogan, and R. Sarikaya, "S-Vector: A Discriminative Representation Derived from i-Vector for Speaker Verification," in *Proc. EUSIPCO*, Nice, France, August 2015.

[21] 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*, Florence, Italy, May 2014.

[22] V. Vasilakakis, S. Cumani, P. Laface, and P. Torino, "Speaker Recognition by means of Deep Belief Networks," in *Proc. Biometric Technologies in Forensic Science*, 2012.

[23] O. Ghahabi and J. Hernando, "Restricted Boltzmann Machine Supervectors for Speaker Recognition," in *Proc. ICASSP*, Brisbane, Australia, April 2015.

[24] P. Safari, O. Ghahabi, and J. Hernando, "From Features to Speaker Vectors by means of Restricted Boltzmann Machine Adaptation," To be appeared in Odyssey Speaker and Language Recognition Workshop, Bilbao, Spain, June, 2016.

[25] G. Hinton, S. Osindero, and Y.-W. Teh, "A Fast Learning Algorithm for Deep Belief Nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[26] G. Hinton, "A Practical Guide to Training Restricted Boltzmann Machines," *Momentum*, vol. 9, no. 1, 2010.

[27] C. Nadeu, D. Macho, and J. Hernando, "Time and Frequency Filtering of Filter-Bank Energies for Robust HMM Speech Recognition," *Speech Communication*, vol. 34, no. 1, pp. 93–114, 2001.

[28] "The NIST Year 2006 Speaker Recognition Evaluation Plan," Tech. Rep., 2006.

[29] A. Larcher, J.-F. Bonastre, B. Fauve, K. Lee, C. Lévy, H. Li, J. Mason, and J.-Y. Parfait, "ALIZE 3.0 - Open Source Toolkit for State-of-the-art Speaker Recognition," in *Proc. Interspeech*, Lyon, France, August 2013.