# RESTRICTED BOLTZMANN MACHINE SUPERVECTORS FOR SPEAKER RECOGNITION

*Omid Ghahabi, Javier Hernando*

TALP Research Center, Department of Signal Theory and Communications
Universitat Politecnica de Catalunya - BarcelonaTech, Spain
{omid.ghahabi, javier.hernando}@upc.edu

## ABSTRACT

The use of Restricted Boltzmann Machines (RBM) is proposed in this paper as a non-linear transformation of GMM supervectors for speaker recognition. It will be shown that the RBM transformation will increase the discrimination power of raw GMM supervectors for speaker recognition. The experimental results on the core test condition of the NIST SRE 2006 corpus show that the proposed RBM supervectors will achieve a comparable performance to i-vectors. Furthermore, the combination of RBM supevectors and i-vectors in the score level improves the performance of the i-vector approach by more than 10% in terms of EER.

***Index Terms***— Speaker Recognition, Supervector, Restricted Boltzmann Machine

## 1. INTRODUCTION

The conventional state-of-the-art method in speaker recognition is the Gaussian Mixture Model-Universal Background Model (GMM-UBM) [1]. In this method, speaker GMM models are adapted from the UBM using maximum a posteriori (MAP) adaptation technique. The main problem of this method is that it is very slow in the testing phase as each frame of the speech signal should be scored separately against both adapted GMM and UBM. Support Vector Machine (SVM) combined with GMM is another successful method in speaker recognition [2]. In this method, the mean vectors of the adapted GMM are concatenated to form a bigger vector called GMM supervector. GMM supervectors are then modeled by the SVM classifier. Supervectors are not only used in speaker recognition (e.g., [2][3]) but are also used in other applications (e.g., [4][5]), which shows the importance of these kinds of features. The most recent state-of-the-art method is well-known as i-vector [6]. Supervectors are transformed to the lower dimensional i-vector space using the effective factor analysis method.

On the other hand, Restricted Boltzmann Machines (RBM) have recently been used in audio and speech processing area (e.g., [7–10]). They were used in speaker recognition for the first time in [11] as unsupervised feature extractors. They were further used in [8][12] to model i-vectors and in [13] to extract speaker factors. In [14] and [9] RBMs have been used to extract pseudo-ivectors from acoustic features and i-vectors, respectively. They have been also employed in an adaptation process to model target and non-target i-vectors discriminatively [15][16][17]. RBMs have been recently used as a pre-training process in Deep Belief Networks (DBN) to extract Baum-Welch statistics for supervector and i-vector extraction [10][18] as well.

In this paper, we use RBMs as a non-linear transformation and dimension reduction stage for GMM supervectors. The normalized version of hidden state likelihoods are considered as RBM supervectors in this paper. Before transformation, supervectors are model-normalized and whitened. It will be shown that the RBM transformation can decrease the dimension of supervectors while increasing their discrimination power. We will show that RBM supervectors achieve comparable performance to i-vectors and their combination in the score level improves the EER by more than 10%.

## 2. GMM SUPERVECTOR AND I-VECTOR

Supervectors are often referred to high dimensional vectors combined of many smaller dimensional ones. In a wider sense, they can be understood as any high- and fixed-dimensional representation of an utterance. GMM supervectors are $(M \times D)$-dimensional vectors obtained by stacking the $D$-dimensional mean vectors of an $M$-mixture adapted GMM. Usually only mean vectors $\boldsymbol{\mu}_i$ are adapted and, therefore, weights $w_i$ and covariance matrices $\boldsymbol{\Sigma}_i$ are the same for both UBM and adapted GMM. Suppose that the UBM, $\lambda_{ubm}$, and the MAP-adapted GMM for speaker $a$, $\lambda_a$, are represented as,

$$\lambda_{ubm} = \left\{w_i, \boldsymbol{\mu}_i^{ubm}, \boldsymbol{\Sigma}_i\right\}_{i=1}^{M}, \tag{1}$$

$$\lambda_a = \left\{w_i, \boldsymbol{\mu}_i^{a}, \boldsymbol{\Sigma}_i\right\}_{i=1}^{M} \tag{2}$$

where $i$ is the index of the $i$th Gaussian mixture, and $\boldsymbol{\Sigma}_i$ is considered diagonal. The supervector $\mathbf{s}^a$ is then represented

This work has been funded by the Spanish project SpeechTech4All (TEC2012-38939-C03-02) and the European project CAMOMILE (PCIN-2013-067).

as,

$$\mathbf{s}^a = (\boldsymbol{\mu}_1^a, \boldsymbol{\mu}_2^a, ..., \boldsymbol{\mu}_M^a)^t \quad (3)$$

where $t$ refers to a transpose operation.

It is assumed that supervectors can be further decomposed as follows [6],

$$\mathbf{s}^a = \mathbf{s}^{ubm} + \mathbf{T}\boldsymbol{\omega} \quad (4)$$

where $\mathbf{T}$ is the low rank total variability matrix which is trained in an iterative process using the centralized Baum-Welch statistics from all available speech utterances. And $\boldsymbol{\omega}$ is a low rank vector referred to as the i-vector. The cosine similarity is an effective distance metric to compare i-vectors when no speaker label is available for development data.

## 3. RBM SUPERVECTOR

Figure 1 shows the block diagram of the process of creating proposed RBM supervectors. Three main techniques are used in our proposed approach, namely model normalization, whitening, and non-linear transformation by RBMs. The objective of the proposed method is to transform conventional raw GMM supervectors to lower dimensional ones while increasing their discrimination power. In comparison to the conventional i-vector technique, the first whitening block and the RBM transformation in Fig. 1 are the main contributions of this work.

### 3.1. Model Normalization

Model normalization has been used in speaker recognition [2]. In that work, MAP adapted mean vectors are normalized by their corresponding UBM weight and variance parameters. In section 4 we will evaluate different combinations of UBM parameter normalization and will conclude that UBM mean and variance normalization will achieves the best performance when the cosine distance is used,

$$\boldsymbol{\mu}_i \leftarrow \boldsymbol{\Sigma}_i^{-1/2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_i^{ubm}), 1 \leqslant i \leqslant M \quad (5)$$

### 3.2. Whitening

A whitening transformation rotates the original data $\mathbf{X}$ to the principle component space in which the rotated data components are uncorrelated and the covariance matrix will ideally be the identity matrix,

$$\mathbf{X}_{whiten} = \mathbf{H}\mathbf{X} \quad (6)$$

$$\mathbf{H} = \mathbf{V}\left(\mathbf{D} + \epsilon\right)^{-1/2}\mathbf{V}^t \quad (7)$$

where $\mathbf{H}$ is the whitening matrix, $\mathbf{V}$ is the matrix of eigenvectors, and $\mathbf{D}$ is the diagonal matrix of the corresponding eigenvalues. And the small constant of $\epsilon$ is added to avoid large values in practice.

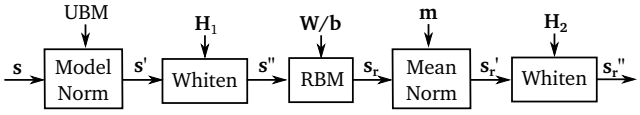


**Fig. 1**: Block-diagram of the process of transformation of raw GMM supervectors ($\mathbf{s}$) to the proposed RBM supervectors ($\mathbf{s}_\mathbf{r}''$). $\mathbf{H}_1$ and $\mathbf{H}_2$ are whitening matrices, $\mathbf{W}$ and $\mathbf{b}$ are RBM parameters obtained on the development data.

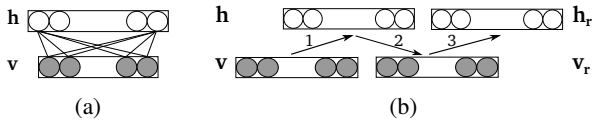


**Fig. 2**: RBM (a) and RBM training (b).

Whitening has been used in speaker recognition in different ways, e.g., as a part of the baseline system in the recent NIST i-vector challenge [19]. Similarly, we whiten the model-normalized supervectors to obtain supervectors with uncorrelated components. It will be shown in section 4 that it plays an important role for increasing the discrimination power of supervectors. Whitening will be used another time in our approach when supervectors are transformed to a lower dimensional space using RBMs.

### 3.3. Restricted Boltzmann Machine

RBMs are generative models composed of two fully connected layers of visible and hidden stochastic units (Fig. 2a). RBMs have been used in speaker recognition for different purposes [8–13] [15–18]. In this paper, we use RBM as a non-linear transform and dimension reduction stage for our normalized supervectors. At first, RBM is trained using development supervectors. Then the trained parameters are used for transforming new supervectors. The inputs to the RBM will be normalized supervectors and the outputs will be hidden state likelihoods computed by eq. 8. Hidden state likelihoods are not suitable as such to be considered as new supervectors. Therefore, we first take their logarithm and then we normalize them. It will be shown in section 4 that mean normalization followed by whitening achieves excellent results.

Training an RBM is based on the Contrastive Divergence (CD) algorithm [20][21]. An approximated version of CD algorithm is called $\text{CD}_1$. From an algorithmic point of view, training an RBM with $\text{CD}_1$ where the input data is real-valued Gaussian distributed can be summarized as follows,

- Initialize Network Parameters ($\mathbf{W}, \mathbf{b}, \mathbf{a}$)
- $\text{CD}_1$ Steps (Fig. 2b)
  1. $\mathbf{h} = \sigma\left(\mathbf{b} + \mathbf{W}\mathbf{v}\right)$ (8)

2. $\mathbf{v}_r = \mathbf{a} + \mathbf{W}^t \mathbf{h}'$ (9)

3. $\mathbf{h}_r = \sigma(\mathbf{b} + \mathbf{W}\mathbf{v}_r)$ (10)

- Update Network Parameters

1. $\Delta\mathbf{W} = \alpha(\mathbf{h}\mathbf{v}^t - \mathbf{h}_r\mathbf{v}_r^t)$ (11)

2. $\Delta\mathbf{a} = \alpha(\mathbf{h} - \mathbf{h}_r)$ (12)

3. $\Delta\mathbf{b} = \alpha(\mathbf{v} - \mathbf{v}_r)$ (13)

$\mathbf{W}$ is the network weight matrix and $\mathbf{a}$ and $\mathbf{b}$ are hidden and visible bias vectors, respectively. Vectors $\mathbf{v}$ and $\mathbf{h}$ are respectively visible and hidden unit values and $\mathbf{v}_r$ and $\mathbf{h}_r$ are their reconstructed ones. The parameter $\alpha$ is the learning rate, $\sigma(x) = (1 + e^{-x})^{-1}$ is the sigmoid function, and $\mathbf{h}'$ is a binary vector randomly sampled from $\mathbf{h}$.

The parameter updating process is iterated until the algorithm converges. It is possible to perform the above parameter update after processing each training example, but it is often more efficient to divide the whole input data (batch) into smaller size batches (minibatch) and update parameters for each minibatch. More details can be found in [20][21][22].

## 4. EXPERIMENTAL RESULTS

### 4.1. Baseline and Database

Features used in the experiments are Frequency Filtering (FF) features [23] extracted every 10 ms using a 30 ms Hamming window. The number of static FF features is 16 and together with delta FF and delta energy, they make 33-dimensional feature vectors. Before feature extraction, speech signals are subjected to an energy-based silence removal process and no feature post-processing is carried out.

The whole core test condition of the NIST 2006 SRE evaluation [24] is used in all experiments. It includes 816 target models and 51,068 trials. Signals have around two minutes of speech. Performance is evaluated using the Equal Error Rate (EER) and the minimum Decision Cost Function (minDCF) calculated using $C_M = 10, C_{FA} = 1$ and $P_T = 0.01$.

The performance of the proposed approach is compared with the i-vector baseline system in which i-vectors are compared using cosine distance. The gender-independent UBM is represented as a diagonal covariance, 512-component GMM. To create supervectors, GMMs are adapted from the UBM by the relevance factor of 16 and 5 EM iterations. Only mean vectors are adapted. ALIZE open source software [25] is used to extract 400-dimensional i-vectors and ($512 \times 33$ = 16,896) - dimensional supervectors.

The development data includes 6,125 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 same development data is used to train UBM, RBM, $\mathbf{T}$ matrix and whitening matrices.

**Table 1**: Comparison of different kinds of supervector model normalization for Euclidean and Cosine distances. Results are obtained on the core test condition of NIST SRE 2006 corpus. $\boldsymbol{\mu}_i^{ubm}$ are UBM mean vectors, $w_i$ and $\boldsymbol{\Sigma}_i$ are respectively weights and diagonal covariance matrices shared between adapted GMMs and UBM.

| **Normalization** | **Distance** | | | |
|---|---|---|---|---|
| | **Euclidean** | | **Cosine** | |
| | EER(%) | DCF | EER(%) | DCF |
| $\boldsymbol{\mu}_i \leftarrow \boldsymbol{\mu}_i$ | 31.09 | 0.0973 | 30.51 | 0.0976 |
| $\boldsymbol{\mu}_i \leftarrow \boldsymbol{\Sigma}_i^{-1/2}\boldsymbol{\mu}_i$ | 30.52 | 0.0979 | 30.23 | 0.0980 |
| $\boldsymbol{\mu}_i \leftarrow \sqrt{w_i}\boldsymbol{\Sigma}_i^{-1/2}\boldsymbol{\mu}_i$ | 29.79 | 0.0971 | 29.47 | 0.0972 |
| $\boldsymbol{\mu}_i \leftarrow (\boldsymbol{\mu}_i - \boldsymbol{\mu}_i^{ubm})$ | 31.09 | 0.0973 | 19.19 | 0.0729 |
| $\boldsymbol{\mu}_i \leftarrow \boldsymbol{\Sigma}_i^{-1/2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_i^{ubm})$ | 30.52 | 0.0979 | **17.69** | **0.0677** |
| $\boldsymbol{\mu}_i \leftarrow \sqrt{w_i}\boldsymbol{\Sigma}_i^{-1/2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_i^{ubm})$ | 29.79 | 0.0971 | 17.82 | 0.0679 |

### 4.2. Results

Table 1 shows the results obtained with different kinds of supervector model normalization for both distance metrics of Euclidean and Cosine. The normalization is carried out by using the UBM mean vectors $\boldsymbol{\mu}_i^{ubm}$, the weights $w_i$, and diagonal covariance matrices $\boldsymbol{\Sigma}_i$ shared between adapted GMMs and UBM. As it was mentioned in section 2, only mean adaptation is carried out. Therefore, the weight and covariance matrices will be the same for UBM and adapted GMMs. The first row of the table compares the two distance metrics when no normalization is considered. As it can be seen, there is only a small difference between these metrics in this case. The second row shows that the variance normalization helps a little bit in both cases. The third row indicates that adding model weights to the normalization improves the results a little bit more. As it was expected, mean normalization does not affect the results in the Euclidean case. However, a big improvement can be observed when mean normalized supervectors are compared using the cosine distance. The variance and mean normalization together show even better performance than using only mean normalization. The last row in the table shows that adding the weights to the normalization process does not help more. It can be concluded from the table 1 that UBM mean and variance normalization increases the discrimination power of supervectors to a great extent when the cosine distance is employed for the similarity measurement.

The resulting supervectors are further whitened to minimize the correlation among supervector components. As it was mentioned in section 3.2, a regularization factor $\epsilon$ is considered in practice to avoid numerical instability in whitening. Fig. 3 shows the variability of EER in terms of this factor. As it can be seen in this figure, there is a good value between 0 and 1 for this parameter which it is equivalent to 0.2 in our application. The minDCF shows also the same behavior.

As it was mentioned in sec. 3.3, normalized supervectors are further transformed by RBM to a lower dimensional space. The objective of this transform is to decrease the dimension of supervectors while increasing their discrimination

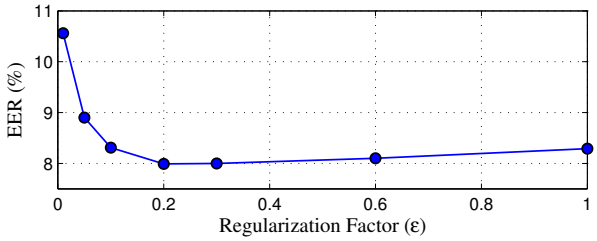


**Fig. 3**: Setting regularization factor in whitening (results are obtained on the UBM Mean-Variance (MV) normalized of raw GMM supervectors).

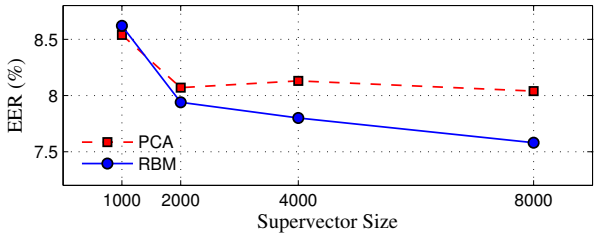


**Fig. 4**: Comparison of PCA and RBM dimension reduction techniques for normalized supervectors in terms of EER.

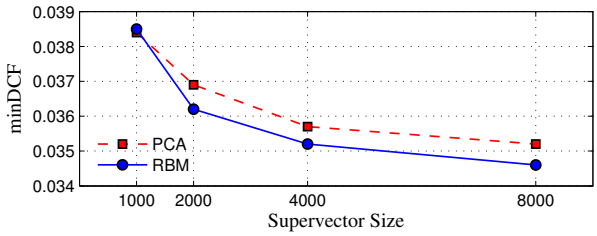


**Fig. 5**: Comparison of PCA and RBM dimension reduction techniques for normalized supervectors in terms of minDCF.

power. Figures 4 and 5 show the efficiency of RBM dimension reduction technique for different hidden layer (or supervector) sizes and compare them with the conventional PCA method. As it was mentioned in sec. 3.3, the logarithm of hidden state likelihoods are mean normalized and then whitened to be suitable as supervectors. As it can be seen in these figures, RBM works similar to PCA for dimensions as low as 1,000 whereas it becomes more efficient than PCA by increasing the dimension of transformed supervectors. The learning rate ($\alpha$), the number of epochs (NofE), and the minibatch size are set respectively to 0.001, 40, and 100 for RBM training. A fixed momentum of 0.9 and a weight decay of $2 \times 10^{-4}$ are also considered.

Table 2 summarizes the effectiveness of each technique used in Fig. 1. As it can be seen in this table, UBM Mean-Variance (MV) normalization can increase the discrimination power of supervectors by 42% in comparison to the raw supervectors. Moreover, the whitening step will decrease the

**Table 2**: The effect of using each technique in the proposed RBM supervector approach of Fig. 1. MV and M Norms stand for Mean-Variance and Mean Normalization, respectively. RBM is used with the hidden layer of 8,000 units. Results are obtained on the core test condition of NIST SRE 2006 corpus.

| **Technique** | **EER(%)** | **minDCF** |
|---|---|---|
| Supervector + Cosine | 30.51 | 0.0976 |
| Supervector + **UBM MV Norm** + Cosine | 17.69 | 0.0677 |
| Supervector + UBM MV Norm + **Whiten** + Cosine | 8.00 | 0.0346 |
| Supervector + UBM MV Norm + Whiten + **RBM + M Norm + Whiten** + Cosine | **7.58** | **0.0346** |

**Table 3**: Comparing the performance of the proposed RBM supervector with the i-vector. RBM is used with the hidden layer of 8,000 units. Results are obtained on the core test condition of NIST SRE 2006 corpus.

| **Technique** | **EER(%)** | **minDCF** |
|---|---|---|
| i-Vector + Cosine | 7.18 | 0.0324 |
| RBM Supervector + Cosine | 7.58 | 0.0346 |
| Combination | **6.45** | **0.0314** |

EER by 55% more. And finally, RBM transform and normalization will decrease the dimension of supervectors by more than an a half while increasing the performance.

Table 3 compares the best result obtained by the proposed supervector with the successful i-vector. In both cases the similarity between identity vectors is measured simply using cosine distance. As it can be seen in this table, the performance of the proposed supervector is comparable with the successful i-vector. Moreover, if these two techniques are combined in the score level, more than 10% improvement will be observed over the baseline i-vector approach. This indicates that the proposed identity supervector gives complementary information to the conventional i-vector. The combination is carried out simply by summing the mean and variance normalized scores.

## 5. CONCLUSION

New lower dimensional supervectors based on Restricted Boltzmann Machines (RBM) are presented in this paper for speaker recognition. Raw GMM supervectors are model-normalized and whitened. Then the resulting supervectors are transformed by RBMs. We have shown that RBM transformation decreases the dimension of supervectors while increases their discrimination power. The experimental results on the core test condition of the NIST SRE 2006 corpus show that the proposed supervectors achieve a comparable performance to i-vectors. Moreover, the combination of the proposed supevectors and i-vectors in the score level improves the performance of the i-vector approach by more than 10% in terms of EER.

## 6. REFERENCES


[1] D.A. Reynolds and R.C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, 1995.

[2] W.M. Campbell, D.E. Sturim, D.A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. ICASSP*, 2006.

[3] R. Dehak, N. Dehak, P. Kenny, and P. Dumouchel, "Linear and non linear kernel gmm supervector machines for speaker verification," in *Proc. Interspeech*, 2007.

[4] T. Bocklet, A. Maier, J.G. Bauer, F. Burkhardt, and E. Noth, "Age and gender recognition for telephone applications based on GMM supervectors and support vector machines," in *Proc. ICASSP*, Mar. 2008, pp. 1605–1608.

[5] X. Zhuang, J. Huang, G. Potamianos, and M. Hasegawa-Johnson, "Acoustic fall detection using gaussian mixture models and GMM supervectors," in *Proc. ICASSP*, Apr. 2009, pp. 69–72.

[6] 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, May 2011.

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

[8] T. Stafylakis, P. Kenny, M. Senoussaoui, and P. Dumouchel, "Preliminary investigation of boltzmann machine classifiers for speaker recognition," in *Proc. Odyssey*, 2012.

[9] S. Novoselov, T. Pekhovsky, K. Simonchik, and A. Shulipa, "RBM-PLDA subsystem for the NIST i-vector challenge," in *Proc. Interspeech*, 2014, pp. 378–382.

[10] W. M. Campbell, "Using deep belief networks for vector-based speaker recognition," in *Proc. Interspeech*, 2014, pp. 676–680.

[11] H. Lee, Y. Largman, P. Pham, and A. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," *Advances in neural information processing systems*, vol. 22, pp. 10961104, 2009.

[12] M. Senoussaoui, N. Dehak, P. Kenny, R. Dehak, and P. Dumouchel, "First attempt of boltzmann machines for speaker verification," in *Proc. Odyssey*, 2012.

[13] T. Stafylakis, P. Kenny, M. Senoussaoui, and P. Dumouchel, "PLDA using gaussian restricted boltzmann machines with application to speaker verification," in *Proc. Interspeech*, 2012.

[14] V. Vasilakakis, S. Cumani, and P. Laface, "Speaker recognition by means of deep belief networks," in *Biometric Technologies in Forensic Science*, 2013.

[15] O. Ghahabi and J. Hernando, "Deep belief networks for i-vector based speaker recognition," in *Proc. ICASSP*, May 2014, pp. 1700–1704.

[16] O. Ghahabi and J. Hernando, "i-vector modeling with deep belief networks for multi-session speaker recognition," in *Proc. Odyssey*, 2014, pp. 305–310.

[17] 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*, Lecture Notes in Artificial Intelligence. Springer, Nov. 2014.

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

[19] "The 2013-2014 speaker recognition i-vector machine learning challenge," 2014.

[20] G.E. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, July 2006.

[21] G.E. Hinton, S. Osindero, and Y-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, May 2006.

[22] G.E. Hinton, "A practical guide to training restricted boltzmann machines," in *Neural Networks: Tricks of the Trade*, number 7700 in Lecture Notes in Computer Science, pp. 599–619. Springer Berlin Heidelberg, Jan. 2012.

[23] 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. 12, pp. 93–114, Apr. 2001.

[24] "The NIST year 2006 speaker recognition evaluation plan," 2006.

[25] A. Larcher, J-F. Bonastre, B. Fauve, K. Lee, C. Lvy, H. Li, J. Mason, and J-Y. Parfait, "ALIZE 3.0 open source toolkit for state-of-the-art speaker recognition," in *Proc. Interspeech*, 2013, pp. 2768–2771.