Universal Background Sparse Coding and Multilayer Bootstrap Network for Speaker Recognition

Xiao-Lei Zhang, Member, IEEE

Abstract—In speaker recognition, Gaussian mixture model based universal background model is a standard for extracting high-dimensional supervectors, and factor-analysis-based i-vector is a recent state-of-the-art method for reducing the high-dimensional supervectors to low-dimensional representations. In this abstract paper, we propose an alternative to the aforementioned techniques by multilayer bootstrap networks (MBN). We first learn a high-dimensional sparse code for each frame by a universal background MBN, and then accumulate the sparse codes of the frames in a session (a.k.a. utterance) into a single high-dimensional sparse supervector. Finally, we reduce the session-level sparse supervectors to a low-dimensional subspace by MBN for unsupervised speaker clustering, or principle component analysis for supervised speaker classification. Our initial result on a small-scale problem demonstrates the effectiveness of the proposed method.

Note that this abstract paper is used to protect the idea. A full version with large-scale experiments will be announced later.

Index Terms—Multilayer bootstrap network, single-layer bootstrap network, density estimation in the discrete space, universal background sparse coding, speaker recognition.

I. INTRODUCTION

GAUSSIAN mixture model (GMM) based universal background model (UBM) [1] is a standard for extracting high-dimensional supervectors in many speech processing tasks, particularly speaker recognition [1]. However, GMM-UBM has the following issues:

- GMM-UBM may make an inaccurate model assumption. It has to make a strong assumption that an underlying data distribution is Gaussian, which may be inaccurate, particularly for a highly-nonlinear data distribution.
- GMM-UBM is not very effective. It generates a localized representation, which is not as effective as a distributed representation when given the same number of hidden states.
- GMM-UBM is complicated. It first optimizes a GMM model by expectation-maximization algorithm, then estimates frame-level posterior probability, and finally gets a supervector of a session (i.e. utterance) by concatenating the accumulation of the mixture occupation of all frames and the vector form of the centered first order statistics. See the open source1 for the standard GMM-UBM model.

In this abstract paper, we propose to extract high-dimensional sparse supervectors by multilayer bootstrap networks (MBN). The proposed simple method, named universal background sparse coding (UBSC), first uses MBN to extract frame-level high-dimensional supervectors, and then accumulates the high-dimensional supervectors of the frames in a session into a single session-level supervector, where the number of nonlinear layers in MBN is set to 1 in most cases (particularly in supervised learning).

We report our initial result in both unsupervised learning and supervised learning. Specifically, (i) for unsupervised speaker clustering, we use UBSC to generate session-level supervectors, and then employ MBN to reduce the dimensionality of the session-level supervectors [2]. The system is called UBSC+MBN. (ii) For supervised speaker classification, we use PCA to reduce the dimension of the output of UBSC, and use linear discriminant analysis (LDA) plus probabilistic LDA (pLDA) to do classification. The proposed system is called UBSC+PCA. Our initial experimental results on a small-scale problem show that UBSC outperforms GMM-UBM in terms of effectiveness, simplicity, and robustness.

II. UNIVERSAL BACKGROUND SPARSE CODING

The proposed UBSC is shown in Fig. 1. Suppose we have \( S \) sessions \( \{\mathcal{U}_s\}_{s=1}^{S} \) with the \( s \)-th session \( \mathcal{U}_s \) defined as

\[
\left( \frac{1}{n} \right) \left( \begin{array}{c}
\mathbf{u}_1 \\
\mathbf{u}_2 \\
\vdots \\
\mathbf{u}_n \\
\end{array} \right)
\]

Frame 1 Frame 2 Frame n Session-level sparse code

Frame-level sparse code

Hidden layer 3 \((k=2)\)

Hidden layer 2 \((k=3)\)

Hidden layer 1 \((k=6)\)

Frame-level acoustic feature

Fig. 1. Principle of UBSC. Each square in MBN represents a \( k \)-centers clustering. The operator “+” denotes element-wise addition between vectors. Note that the number of nonlinear layers of MBN for UBSC is set to 1 in most cases (particularly in unsupervised learning).

1The source code is downloadable from http://research.microsoft.com/en-us/downloads/a6262fec-03a7-4060-a08c-0b0d037a3f5b/
\( \{ x_{s,i} \}_{i=1}^{n_s} \) where \( x_{s,i} \) is the acoustic feature of the \( i \)-th frame of \( \mathcal{U}_s \). UBSC executes the following steps:

- The first step mixes all sessions into a large corpus \( \mathcal{X} = \{ x_i \}_{i=1}^{N} \), where \( N = \sum_{s=1}^{S} n_s \).
- The second step trains an MBN with \( \mathcal{X} \), and generates a \( D \)-dimensional sparse vector \( y_i \) for each frame \( x_i \). Note that, different from [4], MBN does not further reduce the high-dimensional feature to a low-dimensional feature by PCA. Note that we only train a one-hidden-layer MBN (or called single-layer bootstrap network (SBN)) in most cases (particularly for supervised learning). This is because that MBN is used to encode the input data space (estimate the density of data) in a nonparametric way, hence training multiple layers will discard small variations of data in an unsupervised manner, which may be undesired for the later stages, e.g. unsupervised clustering by MBN or supervised classification by LDA.
- The third step generates session-level supervectors \( \{ y_s \}_{s=1}^{S} \) by conducting an element-wise average over the frames that belong to the same session:

\[
\bar{y}_{s,d} = \frac{1}{n_s} \sum_{i=1}^{n_s} y_{s,i,d}, \quad \forall d = 1, \ldots, D
\]

(1)

where \( y_{s,i} = [y_{s,i,1}, \ldots, y_{s,i,D}]^T \) and \( \bar{y}_s = [\bar{y}_{s,1}, \ldots, \bar{y}_{s,D}]^T \).

**A. A typical hyperparameter setting**

The hyperparameter setting of MBN in UBSC follows the suggestion in [2] except that \( k_L \), i.e. hyperparameter \( k \) at the top layer of MBN, is set to a value that is significantly larger than the number of underlying speakers \( c \), e.g. \( k_L = 10c \).

**B. Contribution of UBSC**

UBSC overcomes the weaknesses of GMM-UBM:

- UBSC does not make any model assumption for approximating a data distribution. Instead, it uses bootstrap resampling to do so. Each bootstrap sample, which is generated from data, correctly approximates the data distribution.
- UBSC generates a distributed representation, where each base representation is a localized representation.
- UBSC is very simple, since (i) MBN itself is very simple, and (ii) the process of producing a session-level sparse representation is simply an element-wise average over frame-level sparse representations.

### III. UNSUPERVISED SPEAKER RECOGNITION SYSTEM

The proposed unsupervised speaker recognition system, which uses UBSC to replace GMM-UBM in [2], is shown in Fig. 2. We do not introduce this simple system anymore.

![Diagram of the unsupervised UBSC+MBN speaker recognition system](image)

**IV. SUPERVISED SPEAKER RECOGNITION SYSTEM**

The proposed supervised speaker recognition/classification system is shown in Fig. 2. In the training stage, it uses single-layer bootstrap network (SBN) to learn frame-level high-dimensional codes of training data and averages the frame-level codes in a session for a session-level feature. Then, it reduces the session-level feature to a low-dimensional code by PCA. Finally, it uses LDA and probabilistic LDA (pLDA) to do supervised classification. Note that the SBN model, PCA bases, and LDA model need to be saved for prediction.

![Diagram of the supervised UBSC+PCA speaker recognition system](image)
V. EXPERIMENTS ON UNSUPERVISED SPEAKER CLUSTERING

A. Experimental setup

We adopted the training corpus of speech separation challenge (SSC) [5], which contains 34 speakers and each speaker consists of 500 clean utterances. We selected the first 10 utterances of the first 10 speakers for evaluation, which amounts to 100 utterances. We set the frame length to 25 milliseconds and frame shift to 10 milliseconds, and extracted a 25-dimensional MFCC feature. Eventually, 100 utterances contain 17,385 frames.

For UBSC, we adopted the typical parameter setting of MBN. Specifically, \( V = 400 \), \( a = 0.5 \), \( r = 0 \), and \( k \) were set to 2000-1000-500-250-125 (see [4] for the detailed definitions of the hyperparameters). For the MBN-based unsupervised speaker recognition, we followed the typical parameter setting of MBN in [2]. Specifically, \( V = 400 \), \( a = 0.5 \), \( r = 0.5 \), and \( k \) were set to 90-45-22. The output of PCA was set to \( \{2, 3, 5, 10, 30, 50\} \) dimensions respectively. We assumed that the number of speakers was known, and used \( k \)-means clustering for clustering the low-dimensional data.

We compared our method with [2] where GMM-UBM [1] was used to generate supervectors. We searched the mixture number of GMM-UBM through \( \{2, 4, 8, 16, 32, 64\} \) and found that setting the mixture number of GMM-UBM to 32 performs the best. Therefore, we reported the result of GMM-UBM with 32 mixtures. The MBN in the UBM-MBN based unsupervised speaker recognition adopted the same hyperparameters as that in the UBSC-MBN based system.

|               | 2-dim | 3-dim | 5-dim | 10-dim | 30-dim | 50-dim |
|---------------|-------|-------|-------|--------|--------|--------|
| UBM+PCA       | 64.11 | 69.45 | 78.86 | 83.43  | 77.94  | 74.66  |
| UBSC+PCA      | 71.23 | 71.03 | 83.97 | 91.56  | 86.40  | 83.93  |
| UBM+MBN       | 89.10 | 89.10 | 92.80 | 92.04  | 91.08  | 92.00  |
| UBSC+MBN      | 93.11 | 91.87 | 93.32 | 89.20  | 88.20  | 92.18  |
TABLE II

EER COMPARISON BETWEEN UBM+i-vector AND UBSC+PCA ON SUPERVISED SPEAKER CLASSIFICATION. THE LOWER THE EER IS, THE BETTER THE PERFORMANCE IS.

| parameter \( k \) (number of mixtures) | 2   | 4   | 8   | 16  | 32  | 64  | 128 | 256 | 512 |
|---------------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| UBM+i-vector                          | 0.99| 0.77| 0.18| 0.15| 0.08| 0.04| 0.00| 0.12| 0.27|
| UBSC+PCA                              | 1.42| 0.28| 0.11| 0.04| 0.03| 0.00| 0.01| 0.00| 0.00|

B. Results

Figs. 4 and 5 give a comparison between GMM-UBM and UBSC in data visualization, when PCA is used as the dimensionality reduction tool. From the figures, we observe that UBSC outperforms GMM-UBM apparently, such as differentiating the speakers with yellow and deep-blue colors. Because GMM-UBM has enough mixtures for modeling the 10 speakers, the only reason for their differences is that the data distributions of speakers are not exactly Gaussian.

Figs. 6 and 7 give a comparison between GMM-UBM and UBSC in data visualization, when MBN is used as the dimensionality reduction tool. From the figures, we observe that UBSC performs at least as equally as GMM-UBM, and UBSC seems producing a smaller within-class variance than GMM-UBM.

Table I lists the comparison result between GMM-UBM and UBSC on unsupervised speaker recognition. From the table, we observe that, (i) when PCA is used, UBSC significantly outperforms GMM-UBM, and (ii) when MBN is used, UBSC performs equally to GMM-UBM.

VI. EXPERIMENTS ON SUPERVISED SPEAKER CLASSIFICATION

A. Experimental setup

We adopted the training corpus of SSC. We used the same 10 speakers as in Section V. For each speaker, we randomly selected 100 utterances, half for training and the other half for test. We extracted features in the same way as in Section V.

We compared our UBSC+PCA method with the state-of-the-art UBM+i-vector method [6]. The comparison method uses GMM-UBM to extract supervectors and then uses factor analysis to do the unsupervised dimensionality reduction. The low-dimensional output codes of the GMM+i-vector system, named i-vectors, is used as the input of a supervised LDA/pLDA classifier.

We searched the mixture number of UBM+i-vector through \( \{2, 4, 8, 16, 32, 64, 128, 256, 512\} \). We set the output dimension of i-vector to 100, which is a conventional setting.

For our UBSC, we trained only one nonlinear layer. Specifically, \( V = 200 \), \( a = 0.5 \), \( r = 0 \), and \( k \) was selected from \( \{2, 4, 8, 16, 32, 64, 128, 256, 512\} \). Note that the choices of \( k \) were the same as the choices of the mixture number of the GMM-UBM in the UBM+i-vector system. The output dimension of PCA was set to 100.

We used the common equal error rate (EER) as the evaluation metric. The lower the EER is, the better the performance is.

B. Results

We ran the comparison methods 10 times and recorded the average performance. The results are listed in Table II. From the table, we find that UBSC+PCA is at least as good as UBSC+PCA in terms of minimum EER, and is more robust than UBM+i-vector in terms of the sensitivity to the number of mixtures (or, parameter \( k \)).

VII. CONCLUSIONS

In this abstract paper, we have proposed an MBN-based sparse coding technique, named universal background sparse coding, to generate session-level high-dimensional supervectors. MBN and its extension UBSC estimate the density of data in the discrete space by the simple data resampling, nearest neighbor optimization, and model ensembling. Based on UBSC, we have proposed two systems: (i) UBSC+MBN for unsupervised speaker clustering and (ii) UBSC+PCA for supervised speaker classification.

Our initial experiments on a small-scale subset of the SSC corpus in the noise-free environment show that (i) UBSC+MBN outperforms UBM+PCA uniformly in the scenario of unsupervised clustering; (ii) UBSC+PCA performs at least as good as the state-of-the-art UBM+i-vector system in the scenario of supervised classification. From the initial experiments, we observe that UBSC has a strong potential to outperform GMM-UBM in terms of effectiveness, robustness to hyperparameter selection, and simplicity.

We conjecture that UBSC can be more powerful in more complicated data sets, which needs a further investigation in the full version. In the future, we will conduct more specific experiments on larger data sets, e.g. NIST SRE 2006, SRE 2008, or SRE 2012, in noise-free or noisy environments.

ACKNOWLEDGMENTS

The author thanks Prof DeLiang Wang for providing the Ohio Computing Center and Dr Ke Hu for helping with the SSC corpus.

REFERENCES

[1] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, “Speaker verification using adapted gaussian mixture models,” Digital Signal Process., vol. 10, no. 1, pp. 19–41, 2000.
[2] X.-L. Zhang, “Multilayer bootstrap network for unsupervised speaker recognition,” arXiv preprint arXiv:1509.06095, 2015.
[3] X. Zhao, Y. Wang, and D. L. Wang, “Cochannel speaker identification in anechoic and reverberant conditions,” IEEE/ACM Trans. Audio, Speech, Lang. Process., vol. 22, no. 11, pp. 1727–1736, 2015.
[4] X.-L. Zhang, “Multilayer bootstrap networks,” arXiv preprint arXiv:1408.0849, pp. 1–41, 2014.
[5] M. Cooke and T.-W. Lee, “Speech separation challenge,” http://staffwww.dcs.shef.ac.uk/people/M.Cooke/SpeechSeparationChallenge.htm, 2006.
[6] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speaker verification,” IEEE Trans. Audio, Speech, Lang. Process., vol. 19, no. 4, pp. 788–798, 2011.