INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Speaker Clustering by Iteratively Finding Discriminative Feature Space and Cluster Labels

Sungrack Yun, Hye Jin Jang, Taesu Kim

Qualcomm Research[†],
119 Nonhyun-dong, Gangnam-gu, Seoul, 135-820, Korea
{sungrack, hyejinj, taesu}@qti.qualcomm.com

Abstract

This paper presents a speaker clustering framework by iteratively performing two stages: a discriminative feature space is obtained given a cluster label set, and the cluster label set is updated using a clustering algorithm given the feature space. In the iterations of two stages, the cluster labels may be different from the true labels, and thus the obtained feature space based on the labels may be inaccurately discriminated. However, by iteratively performing above two stages, more accurate cluster labels and more discriminative feature space can be obtained, and finally they are converged. In this research, the linear discriminant analysis is used for discriminating the i-vector feature space, and the variational Bayesian expectation-maximization on Gaussian mixture model is used for clustering the i-vectors. Our iterative clustering framework was evaluated using the database of keyword utterances and compared with the recently-published approaches. In all experiments, the results show that our framework outperforms the other approaches and converges in a few iterations.

Index Terms: speaker clustering, iterative framework, linear discriminant analysis, i-vector, variational Bayesian EM

1. Introduction

With the recent explosive developments of mobile and IoT devices, voice command interfaces [1, 2, 3] have been getting increasing attention and widely adopted on various devices due to its convenience and intuitiveness. In addition, the speaker's voice identification and verification [4, 5, 6] are becoming attractive features for user-specific services. To provide such services, speaker clustering [7, 8] plays a key role in identifying the number of speakers and grouping the utterances from the same user for the automatic user-specific model generation or speaker diarization [9, 10].

A number of researches on speaker clustering and diarization have been proposed [7, 11, 12, 13]. In [12], without any hand-crafted feature, the presented system performs clustering and diarization using a speaker separation deep neural network (DNN) and the derived DNN adapted to input audio. In [11], variational Bayesian expectation-maximization (VBEM) on Gaussian mixture model (GMM) is used for an unsupervised speaker diarization given the i-vector [14] projected by principal component analysis (PCA). In [7] and [13], a multilayer bootstrap network (MBN) with k-means clustering and an artificial neural net with GMM/hidden Markov model are used for speaker clustering and diarization, respectively.

This paper presents a speaker clustering framework by iteratively performing two stages: a discriminative feature space is obtained given a cluster label set, and the cluster label set is updated using a clustering algorithm on the feature space. In contrast to the previous researches [7, 13] where feature extraction and speaker clustering are separately performed, our clustering framework jointly obtains the discriminative feature space and cluster labels by iteratively performing both stages until convergence is met. The block-diagram of our framework is illustrated in Fig. 1. In the first stage, by utilizing the cluster labels from the previous iteration, a feature space is obtained such that the distances between within-cluster samples are minimized while the distances between inter-cluster samples are maximized. In this research, the i-vector representation followed by linear discriminant analysis (LDA) is adopted since this combination has shown promising results and have been successfully applied to speaker verification [4], speaker recognition [5], and language identification [15]. In the second stage, the cluster label set is updated by performing the clustering on the feature space obtained in the first stage. Then, the updated cluster label set is passed to the first stage for the next iteration. In this research, the VBEM-GMM [11, 16] is used for the clustering algorithm.

In the initial iterations of the considered framework, some cluster labels may be incorrectly assigned. However, the correctly assigned labels help to obtain the feature space more discriminatively. By performing several iterations, the cluster label set and the feature space become more accurate and discriminative, respectively, and finally they converge. This is different from the works in [11, 7] where the feature space is obtained in an unsupervised manner using PCA and MBN, respectively. Our clustering framework was evaluated and compared with [11, 7] using the CHiME 2013 database [17]. In all experiments, the results show that our framework outperforms previous works.

2. Speaker Clustering

Speaker clustering is the task of partitioning a set of N observations, $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$, into K disjoint clusters such that the observations in a cluster correspond to the same speaker. The clustering assigns a label set $\mathbf{Y} = \{y_1, ..., y_N\}$ to $\mathbf{X}$, and $y_i \in \{1, ..., K\}$. Each observation $\mathbf{x}_i$ of dimension D can be a speech utterance itself or an encoded vector using various feature extraction algorithms for speaker clustering such as Mel frequency cepstral coefficients (MFCCs) [18], glottal to noise excitation ratio (GNE) [8], i-vector [11, 14, 10, 8] and MBN [7]. In this paper, we use the i-vector for the feature vector of an observation, and it can be obtained by:

$$M = m + Tw$$

where M is the speaker- and session-dependent supervector representing a speaker utterance, m is the speaker- and session-independent supervector obtained from universal background

[†]Qualcomm Research is a division of Qualcomm Technologies, Inc.

http://dx.doi.org/10.21437/Interspeech.2017-923

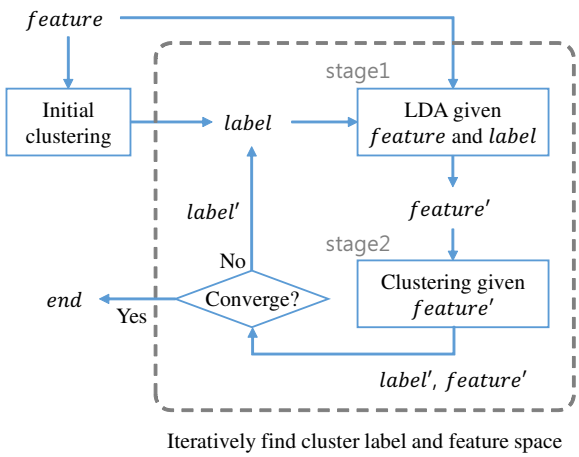

Figure 1: *Proposed clustering framework. In stage1, a discriminative feature space is obtained from the label set obtained in the stage2 of previous iteration. In stage2, the label set is updated by the clustering on the feature space. Two stages are performed iteratively until feature space or label set converges.*

model (UBM), T is total variability matrix containing the speaker and channel variabilities, and w is the i-vector whose components are the total factors [14].

The extracted vectors can be clustered using various clustering algorithms. Especially, the clustering should be performed with an unknown number of clusters for the applications such as speaker diarization and automatic user model generation on IoT devices as described in Sec. 1. There exist a variety of previous approaches applied on the tasks when K is unknown such as bottom-up hierarchical clustering using the Bayesian information criterion (BIC) [19, 20] and top-down hierarchical clustering using the Dirichlet process on hidden Markov model [21, 22]. Also, in [23], the spectral clustering is used to estimate the number of speakers. In this research, we adopt the clustering algorithm, VBEM-GMM [11, 16], to simultaneously perform the clustering and obtain the number of speakers, K. Based on this, the proposed framework also finds the discriminative feature space by utilizing the output of VBEM-GMM, and an iterative process is applied until the cluster label or the feature space converges. In the next section, we will briefly review the VBEM-GMM and then explain the proposed iterative clustering framework.

2.1. VBEM-GMM Clustering

Suppose that the observations $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ are sampled from an mixture of Gaussian distribution. That is, there is a label set $\mathbf{Y} = \{y_1, \cdots, y_N\}$, $y_i \in \{1, ..., K\}$, such that $p(\mathbf{x}_i|y_i = k)$ follows the normal distribution with mean μ_k and variance Σ_k. Regarding $\theta := \{\pi, \mu, \mathbf{\Sigma}, \mathbf{Y}\}$ as random variables, the goal of the VBEM-GMM is to find the distribution $q(\theta)$ which approximates the posterior distribution $p(\theta|\mathbf{X})$ with respect to the KL-divergence under the certain restriction: $q(\theta)$ can be factorized as $q(\theta) = q(\mathbf{Y})q(\pi, \mu, \mathbf{\Sigma})$. It is known that such a $q(\theta)$ also maximizes the lower bound of $p(\mathbf{X})$:

$$\mathcal{L}(q) = \int d\theta\, q(\theta) \ln \frac{p(\mathbf{X}, \theta)}{q(\theta)}.$$

The VBEM-GMM is a process to maximize $\mathcal{L}(q)$ locally with respect to $q(\mathbf{Y})$ and $q(\pi, \mu, \mathbf{\Sigma})$ iteratively. The iterative procedure provides an algorithmic way of finding $q(\theta)$ locally maximizing $\mathcal{L}(q)$. More detailed explanations are described in [16, Section 10.2], [24].

One virtue of the VBEM-GMM is that the number of clusters can be learned. Starting from a large K_M, we can guess the number of clusters K by discarding "empty" clusters, i.e., clusters with the expectation $\mathbb{E}(\pi_k) \simeq 0$.

Also, the distribution $q(\mathbf{Y})$ gives the probability of each observation belonging to each cluster. The labeling of each observation for the following the LDA step can be obtained by choosing the cluster with highest probability.

2.2. Iterative Speaker Clustering Framework

As illustrated in Fig. 1, the proposed clustering framework iteratively finds the discriminative feature space in `stage1` and the cluster labels in `stage2` until one of them converges. The pseudo code for this algorithm is shown in Algorithm 1. First, the initial value $(\mathbf{X}_0, \mathbf{Y}_0)$ are obtained by extracting i-vectors from speech observations $\mathbf{X}$ and clustering the i-vectors using VBEM-GMM, respectively. In clustering $\mathbf{X}_0$ using VBEM-GMM, the initial cluster number is set to K_M which is a preset value denoting the maximum possible number of clusters. In the first stage of i-th iteration (line 7-8 of Algorithm 1), the next feature space $\mathbf{X}_{i+1}$ is obtained by LDA given $(\mathbf{X}_0, \mathbf{Y}_i)$, and the number of unique cluster labels of $\mathbf{Y}_i$, denoted by $U(\mathbf{Y}_i)$, is assigned to K_{i+1} which will be used as the initial cluster number of the next VBEM-GMM clustering. Note that in the iteration loop, the input feature of LDA does not change but is fixed to $\mathbf{X}_0$. The ouput of LDA is changed only by $\mathbf{Y}_i$. In the second stage (line 11 of Algorithm 1), given the updated feature space $\mathbf{X}_{i+1}$ and the number of clusters K_{i+1}, the VBEM-GMM outputs $\mathbf{Y}_{i+1}$. These two stages are performed iteratively until one of the following convergence conditions is met: the difference between the previous feature space and the current one is less than a preset value ϵ, or the cluster labels of the previous and current iteration are the same. When none of them meet the condition, the iteration can be performed I_T times. In Fig. 2, one example of the iterative clustering is shown. The number of clusters started from seven and converged to three in several iterations.

Algorithm 1 Interative Speaker Clustering Algorithm

1: **procedure** ITERATIVE-VBEM-LDA-CLUSTERING($\mathbf{X}$)
2: $\quad\mathbf{X}_0 \leftarrow$ i-vector extracted from $\mathbf{X}$
3: $\quad K_0 \leftarrow K_M$
4: $\quad\mathbf{Y}_0 \leftarrow$ VBEM-GMM$(\mathbf{X}_0, K_0)$
5: $\quad$**for** $i = 0 \rightarrow I_T$ **do**
6: $\quad\quad$// `stage1` *: find discriminative feature space*
7: $\quad\quad\mathbf{X}_{i+1} \leftarrow$ LDA$(\mathbf{X}_0, \mathbf{Y}_i)$
8: $\quad\quad K_{i+1} \leftarrow U(\mathbf{Y}_i)$
9:
10: $\quad\quad$// `stage2` *: find cluster labels*
11: $\quad\quad\mathbf{Y}_{i+1} \leftarrow$ VBEM-GMM$(\mathbf{X}_{i+1}, K_{i+1})$
12:
13: $\quad\quad$**if** $|\mathbf{X}_{i+1} - \mathbf{X}_i| < \epsilon$ or $\mathbf{Y}_{i+1} = \mathbf{Y}_i$ **then**
14: $\quad\quad\quad$break
15: $\quad\quad$**end if**
16: $\quad$**end for**
17: **end procedure**

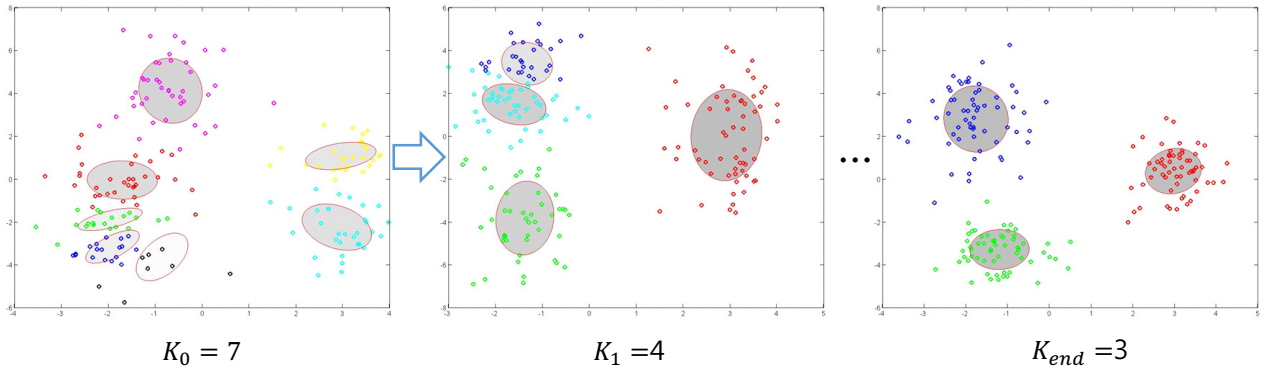

Figure 2: *One example of proposed clustering framework. Here, the points with the same color correspond to the same cluster. The ellipse size and its shade intensity represent GMM variances and responsibilities, respectively. In the initial iteration shown in the left figure, the labels with 7 clusters are obtained. In the next iteration shown in the middle figure, a feature space is discriminated based on the label, and clustering is performed on the feature space. Note that the feature point is changed and different from the initial iteration. After a number of iterations, finally, we obtain a converged space and label set shown in the right figure. In the final iteration, the feature space becomes more discriminative than that in the initial iteration.*

2.2.1. *The clustering with unknown K*

For the fast convergence of the framework, we add the *dimension cutout process* which removes the dimensions with small eigenvalues of LDA. Usually, the number of clusters is larger than the true value before the iteration reaches the converge point. Thus, in the middle of iterations, the feature space dimensions is also larger than the desired one since it is calculated as the number of clusters minus one in the LDA. Due to the projection on a larger dimension space, not all eigenvalues of LDA are dominant: some of them have significant values while others do not. So, we remove the dimensions with small eigenvalues by the following procedure:

1. Find $D' \in \{1, ..., D\}$ such that $\sum_{d=1}^{D'} \lambda_d \geq C$.
2. Obtain the feature space generated by eigenvectors corresponding to λ_d, $d = 1, \cdots, D'$.

where λ_d is the eigenvalue sorted in descending order. The threshold C is a preset value and can be determined differently according to the dimension, i.e. $C(D')$. With this process, we can speed up the iterative process and reduce the convergence time.

2.2.2. *The clustering with known K*

In this case, we assume that the number of clusters is known in the *dimension cutout process*, and thus D' is set to $K - 1$. This setup can serve as a baseline performance to check how accurately D' is estimated in the setup of Sec. 2.2.1, and finally how much the performance is affected by D'.

3. Experiment

We evaluated our clustering framework using the CHiME 2013 database [17] which was created for the 2nd speech separation and recognition challenge with two tracks. In this experiment, we chose the track1 database consisting of keyword utterances from 34 speakers. From each utterance, we extracted 13 dimensional MFCCs with delta and delta-delta coefficients. The frame size and rate was set to 25ms and 10ms, respectively. Speaker i-vectors were extracted from MFCCs with the UBM of 64 mixtures.

The database was split into the training set of 20 speakers (11 male and 9 female) and evaluation set of 14 speakers (7 male and 7 female) without any speaker overlap. We obtained the UBM using the training set and performed clustering using the evaluation set. Each utterance in the dataset contains a sequence of 6 words: command, color, preposition, letter, number and adverb. For a short keyword clustering, we used only the first two words by segmenting the utterances using the word boundary label which is included in the database. With this segmentation, we have the utterances of 16 different keywords: 4 types of commands (*bin*, *lay*, *place*, *set*) followed by 4 types of colors (*blue*, *green*, *white*, *red*).

For the evaluation, we generated a number of clustering datasets by selecting speakers variously: the combinations of selecting $K \in \{2, 3, 4, 5, 7\}$ speakers from all 14 speakers in the evaluation set. For $K = 2$ and $K = 3$, we generated all combinations and obtained $C_2^{14} = 91$ and $C_3^{14} = 364$ clustering datasets, respectively. For $K \in \{4, 5, 7\}$, we only generated 500 clustering datasets since there are too many possible combinations. Also, in generating the clustering datasets, we added noisy utterances by artificially mixing the original clean utterances with a babble and car noise sound for the applications performed in such noisy environments: IoT home devices or intelligent vehicles.

For each speaker, we have 500 utterances with 16 different keywords. In the first experiment, we randomly chose 20 from 500 utterances so that the clustering dataset contains different keywords. In the second experiment, we chose the 20 utterances with the same keyword: the keyword '*bin blue*' was chosen. In evaluating our clustering framework, due to the local minima problem of EM, each clustering was performed 10 times, and the average results across all trials were computed.

Given the generated clustering dataset, we evaluated four different clustering frameworks: k-means clustering given MBN features (Kmeans-MBN), VBEM-GMM clustering given the i-vectors projected by PCA (VBEM-PCA), and the proposed iterative clustering framework when K is unknown (iVBEM-LDA) and known (iVBEM-LDA-K). The Kmeans-MBN presented recently in [7] separately finds the feature space using the MBN and cluster labels using k-means. The VBEM-

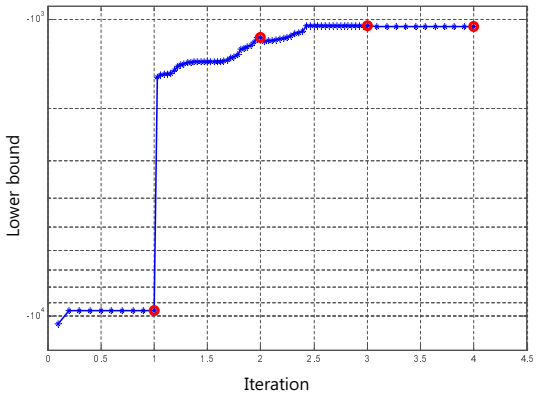

Figure 3: *One example of $\mathcal{L}$ versus iteration. The red circles display the lower bound values when the LDA projection is applied. Between the LDA projections, the VBEM clustering is performed with a number of iterations. In this example, the lower bound is almost monotonically increased, and the lower bound in the converge point is about 10 times greater than that in the initial point.*

PCA was used in [11] for speaker clustering as well as the speaker segmentation for diarization. In this experiment, we used the VBEM-PCA only for clustering. Since the feature space projected by PCA is not affected by labels, it does not need to apply the proposed iterative process: no further iteration is performed. We chose 10 and 5 dimension for the initial i-vector and PCA projection, respectively. In the clustering using iVBEM-LDA and iVBEM-LDA-K, we set $K_M = 7$ and $I_T = 50$. For the performance metric in the evaluation, we used the average cluster purity (ACP) [25].

In the first experiment, given the dataset of random keywords, we obtained the ACPs of various clustering frameworks as summarized in Table 1. The first row of the table indicates the number of speakers K in the clustering dataset. The Kmeans-MBN and iVBEM-LDA-K were performed with known K, and VBEM-PCA and iVBEM-LDA were performed with unknown K. For all cases, the iVBEM-LDA and iVBEM-LDA-K outperform Kmeans-MBN by 5.13% and 18.68% ACP in average, respectively. Even though the clustering was performed with unknown K, the iVBEM-LDA shows better ACP than the Kmeans-MBN. Also, the performance of iVBEM-LDA-K is much better than the iVBEM-LDA especially for the large number of speakers: $K = 5$ and 7. The Fig. 3 shows the lower bound $\mathcal{L}$ as iteration goes on: in this case, the iVBEM-LDA converges with only 4 iterative steps. In this figure, $\mathcal{L}$ increases almost monotonically and converges to the point 10 times greater than the initial point. The iVBEM-LDA and iVBEM-LDA-K perform much better than the VBEM-PCA. In contrast to the VBEM-PCA where the dimension is fixed, our framework actively changes the feature space dimensions by the iterative process of LDA and clustering, and this helps to find the more accurate number of clusters and the labels.

Table 1: *The ACP(%) of four clustering frameworks given the dataset of random keywords in CHiME 2013 database.*

	2	3	4	5	7
Kmeans-MBN	88.85	71.48	67.16	55.81	50.51
VBEM-PCA	31.59	41.05	45.44	46.70	43.31
iVBEM-LDA	91.82	77.32	77.27	62.30	50.73
iVBEM-LDA-K	**93.24**	**92.04**	**89.42**	**84.05**	**68.47**

Table 2: *The ACP(%) of four clustering frameworks given the dataset of the same keyword in CHiME 2013 database.*

	2	3	4	5	7
Kmeans-MBN	98.05	95.52	96.75	96.27	96.76
VBEM-PCA	31.86	49.07	65.14	74.65	79.12
iVBEM-LDA	92.94	97.63	99.94	99.08	95.57
iVBEM-LDA-K	**99.70**	**99.89**	**99.98**	**99.96**	**97.56**

We also evaluated the clustering frameworks using the metric, normalized mutual information, used in [7]. However, the performance of Kmeans-MBN was below that reported in [7] since we used a different experiment setup. In [7], the first 100 utterances of each speaker were chosen, and thus many utterances with the same keyword were included in the clustering dataset due to the database file order. However, in this experiment, we selected the utterances randomly, and most utterances have different keywords. Also, we only took the first two keywords by segmentation and added the noisy utterances in the dataset. Thus, the performance number becomes lower.

In the second experiment, we performed the clustering given only clean utterances of the same keyword. The results are summarized in Table 2, the ACP values are quite higher than those in the first experiment due to the same keyword without noisy utterances. Also, in this experiment, the iVBEM-LDA and iVBEM-LDA-K outperform Kmeans-MBN and VBEM-PCA for all cases except the Kmeans-MBN when $K = 2$ and 7. Compared with the Kmeans-MBN, the iVBEM-LDA and iVBEM-LDA-K show 0.36% and 2.75% ACP improvement in average, respectively. From all results, we can conclude that the clustering by jointly finding feature space and labels consistently shows better performance than the Kmeans-MBN which separately finds the feature space and the labels.

4. Conclusions and Further Work

In this paper, we considered a clustering framework where a discriminative feature space and cluster labels are jointly obtained by an iterative process. In the first stage, LDA is applied on the i-vectors of speech observations given the cluster labels obtained in the previous iteration. In the second stage, the labels are updated using the VBEM-GMM clustering on the obtained feature space. Two stages are iteratively performed until the labels or the feature space converges. We evaluated our clustering framework using the CHiME 2013 track1 database which contains the utterances of six-keyword sequences from 34 speakers. The performance was compared with the algorithms which do not find the feature space and cluster labels jointly. The experimental results show that our clustering framework performs better than the others especially when the clustering dataset consists of the utterances with different keywords. For the further work, we will consider a better approach for the *dimension cutout process* to reduce the performance gap between iVBEM-LDA and iVBEM-LDA-K.

5. References

[1] C. L. Ortiz, "The road to natural conversational speech interfaces," *IEEE Internet Computing*, vol. 18, no. 2, pp. 74–78, 2014.

[2] M. Katore and M. R. Bachute, "Speech based human machine interaction system for home automation," in *Proceedings of IEEE Bombay Section Symposium (IBSS)*, 2015, pp. 1–6.

[3] M. Majewski and W. Kacalak, "Human-machine speech-based interfaces with augmented reality and interactive systems for controlling mobile cranes," in *Proceedings of International Conference on Interactive Collaborative Robotics*, 2016, pp. 89–98.

[4] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Process*, 2014, pp. 4052–4056.

[5] D. Garcia-Romero and A. McCree, "Supervised domain adaptation for i-vector based speaker recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Process*, 2014, pp. 4047–4051.

[6] C. Zhang, S. Ranjan, M. K. Nandwana, Q. Zhang, A. Misra, G. Liu, F. Kelly, and J. H. Hansen, "Joint information from nonlinear and linear features for spoofing detection: an i-vector/DNN based approach," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Process*, 2016, pp. 5035–5039.

[7] X.-L. Zhang, "Universal background sparse coding and multilayer bootstrap network for speaker clustering," in *Proceedings of the INTERSPEECH*, 2016, pp. 1858–1862.

[8] A. Woubie, J. Luque, and J. Hernando, "Improving i-vector and PLDA based speaker clustering with long-term features," in *Proceedings of the INTERSPEECH*, 2016, pp. 372–376.

[9] S. H. Yella and H. Bourlard, "Overlapping speech detection using long-term conversational features for speaker diarization in meeting room conversations," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 22, no. 12, pp. 1688–1700, 2014.

[10] G. Sell and D. Garcia-Romero, "Speaker diarization with PLDA i-vector scoring and unsupervised calibration," in *Proceedings of IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 413–417.

[11] S. H. Shum, N. Dehak, R. Dehak, and J. R. Glass, "Unsupervised methods for speaker diarization: An integrated and iterative approach," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 21, pp. 2015–2028, 2013.

[12] R. Milner and T. Hain, "DNN-based speaker clustering for speaker diarisation," in *Proceedings of the INTERSPEECH*, 2016, pp. 2185–2189.

[13] S. H. Yella and A. Stolcke, "A comparison of neural network feature transforms for speaker diarization," in *Proceedings of the INTERSPEECH*, 2015, pp. 3026–3030.

[14] N. Dehak, P. J. 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.

[15] D. Martínez, L. Burget, L. Ferrer, and N. Scheffer, "iVector-based prosodic system for language identification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Process*, 2012, pp. 4861–4864.

[16] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, USA, Springer, 2006.

[17] E. Vincent, J. Barker, S. Watanabe, J. Le Roux, F. Nesta, and M. Matassoni, "The second CHiME speech separation and recognition challenge: Datasets, tasks and baselines," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Process*, 2013, pp. 126–130.

[18] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, 2012.

[19] D. A. Reynolds and P. Torres-Carrasquillo, "The mit lincoln laboratory rt-04f diarization systems: Applications to broadcast audio and telephone conversations," in *in Proceedings of NIST Rich Transcript. Workshop*, 2004.

[20] F. Valente and C. Wellekens, "Variational bayesian methods for audio indexing," in *International Workshop on Machine Learning for Multimodal Interaction*, 2005, pp. 307–319.

[21] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, "A sticky hdp-hmm with application to speaker diarization," *The Annals of Applied Statistics*, pp. 1020–1056, 2011.

[22] M. J. Johnson and A. Willsky, "The hierarchical dirichlet process hidden semi-markov model," in *Proceedings of Conference on Uncertainty in Artificial Intelligence*, 2010.

[23] S. Shum, N. Dehak, and J. Glass, "On the use of spectral and iterative methods for speaker diarization," in *Proceedings of the INTERSPEECH*, 2012.

[24] M. J. Beal, *Variational algorithms for approximate Bayesian inference*. Ph. D. dissertation, Univ. College London, London, U. K., 2003.

[25] C. D. Manning, P. Raghavan, H. Schütze *et al.*, *Introduction to information retrieval*. Cambridge university press Cambridge, 2008, vol. 1, no. 1.