Department of Electrical Engineering

EE546
Mathematics of High-Dimensional Data
Final Project Report

Spectral Clustering for Speaker Diarization Following the x-vector/PLDA Paradigm

Nikolaos Flemotomos
USC I.D.: 7149389176

December 3, 2018

Abstract

Speaker diarization, the problem of finding "who spoke when" in a speech document, plays a crucial role in numerous applications and lots of research efforts have been focused on it over the last few years. In this work, we are trying to incorporate spectral clustering into the pipeline of the state-of-the-art approach which is followed to tackle the problem. We are experimenting with different system designs and we are evaluating our method on a standard dataset used in the field.

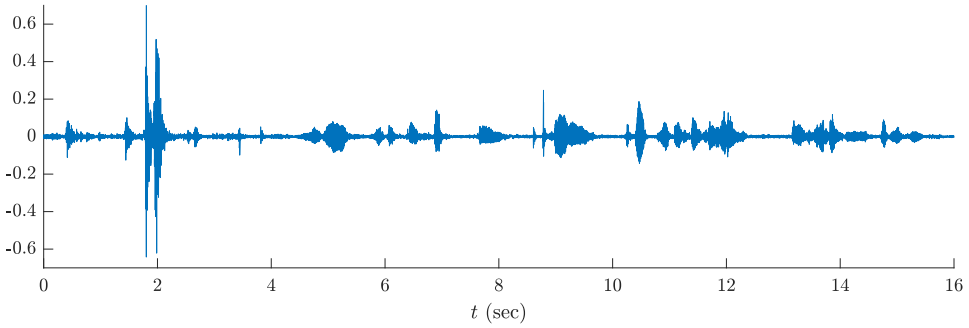
1 Introduction

Given a speech signal with multiple speakers talking, speaker diarization answers the question "who spoke when". Conceptually, we can think of speaker diarization as encompassing three main stages [1, 2]. First, a Speech Activity Detection (SAD)[1] module detects speech vs. non-speech (e.g. silence or noise) regions in the input signal. Then, inside each speech region, a speaker segmentation module finds the speaker change points, that is the timestamps when there is a transition from a particular speaker to another one. That way, we get a collection of speaker-homogeneous segments. Finally, a speaker clustering module clusters those segments into same-speaker groups. The job of a speaker diarization system is visually depicted in Figure 1.

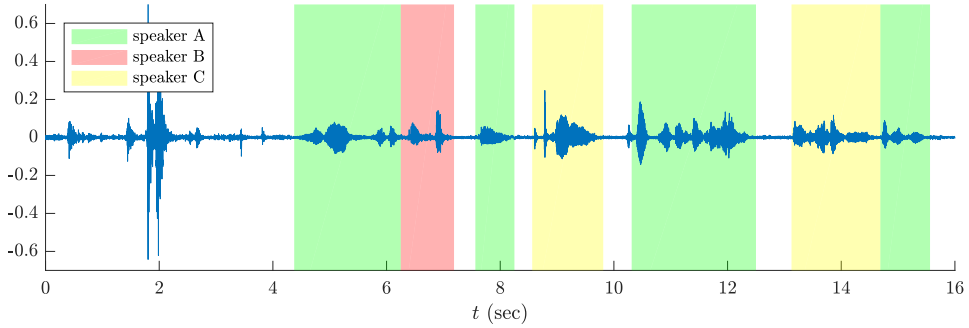
Speaker diarization is a task of utmost importance in speech processing, since it is helpful for other applications such as speech recognition, speaker identification, automatic summarization, etc. This is why it has gained significant popularity over the last years with numerous research efforts trying to tackle the various challenges related to the problem. The traditional approach has been to extract the Mel-Frequency Cepstrum Coefficients (MFCCs), which are fixed-dimensional (in the range 10-40) feature vectors representing the spectral characteristics of a 20-30msec window, model speech segments under some probability distribution (e.g. Gaussian Mixture Models - GMMs), and measure the distance between consecutive segments using some metric, such as the metric based on the Bayesian Information Criterion (BIC), the Generalized Likelihood Ratio (GLR), or the Kullback-Leibler divergence (KL or KL2) [1, 2]. When the distance between two consecutive speech segments is large, a speaker change point has been detected. After this step, the speaker-homogeneous segments are grouped together using Hierarchical Agglomerative Clustering (HAC) (bottom-up approach where initially all the segments are assumed to belong to different speakers), using again a similar distance metric.

More recently, speaker modelling by GMMs has been replaced by i-vectors [3], which are embeddings based on the total variability model. In this framework, the cosine distance metric was initially proposed as the divergence criterion to be used, but Probabilistic Linear Discriminant Analysis (PLDA) -based scoring has been proved to yield improved results [4]. With the advent of Deep Neural Networks (DNNs), there has been an increasing interest to apply deep learning techniques for the task in hand. Indeed, replacing i-vectors by neural embeddings, sometimes called x-vectors [5, 6], has led to advanced performance for the task of speaker diarization and is now the state-of-the-art approach. In this work, we are trying to incorporate spectral clustering [7], which has been successfully applied to various applications, into this framework.

[1] SAD is by itself an entire area of research.



(a) Raw signal.



(b) Diarization output.

Figure 1: Finding "who spoke when" in a speech signal. In (b), the white regions indicate silence or noise. The 5 detected (colored) speech regions are further segmented into 7 speaker-homogeneous segments which are clustered into 3 same-speaker groups.

The rest of the report is structured as follows: Section 2 reviews the key ideas behind spectral clustering and presents the main work that has been done using this technique in the field of speaker diarization. Section 3 reviews a state-of-the-art approach for diarization, that is the x-vector/PLDA framework. Section 4 introduces our approach for using spectral clustering for speaker diarization, with the corresponding experiments presented in Section 5. Finally, Section 6 gives a summary of the presented approach and results.

2 Spectral Clustering and Diarization - Previous Work

Let N data points, which for our application are speech segments. Once we have computed all the pairwise similarities and thus constructed an $N \times N$ affinity matrix $\mathbf{W}$, which can be though of as the adjacency matrix of a weighted graph, spectral clustering [7] exploits the eigenvalue/eigenvector characteristics of $\mathbf{W}$. Assuming $\mathbf{W}$ has only non-negative entries, then the spectral clustering algorithm proceeds as follows:

1. Define the degrees $d_i = \sum_j \mathbf{W}_{ij}$ and the diagonal matrix $\mathbf{D} = \text{diag}\{d_1, d_2, \cdots\}$.

2. Construct the normalized Laplacian $\mathbf{L} = \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2}$.

3. Find the k eigenvectors $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_k$ of $\mathbf{L}$ corresponding to the k largest eigenvalues, where k is the desired number of clusters. Form the matrix $\mathbf{X} = [\mathbf{x}_1 | \mathbf{x}_2 | \cdots | \mathbf{x}_k]$.

4. Normalize the rows of $\mathbf{X}$ and form $\mathbf{Y}$, where $\mathbf{Y}_{ij} = \mathbf{X}_{ij} / \sqrt{\sum_j \mathbf{X}_{ij}^2}$

5. Cluster the N rows of $\mathbf{Y}$ and assign the original i-th point to cluster j if and only if the i-th row of $\mathbf{Y}$ is assigned to cluster j.

Because of the proven theoretical guarantees of the algorithm [7], as well as the various domains to which it has been succesfully applied, there have been several attempts of applying spectral clustering to the problem of speaker diarization. The first relevant work in the field [8] applies spectral clustering to build a speaker segmentation system based on the timing differences in multichannel audio signals. In [9] the authors build the affinity matrix using the KL divergence and replace HAC by spectral clustering achieving similar performance with lower computational complexity. The number of clusters is chosen based on the "eigengap criterion", by searching a drastic drop (or equivalently a drastic increase) in the magnitude of the eigenvalues of the Laplacian matrix. The eigengap criterion is also used in [10], [11], and [12] to determine the number of clusters. Spectral clustering is used in conjunction with i-vectors and cosine distance in [11]. However, it is shown that when the number of speakers is given, spectral clustering gives worse results, compared to k-means on the original i-vectors. In [13], two cluster criterion functions are proposed that maximize the separation between intra-cluster and inter-cluster distances. Those functions are applied in the spetral subspace, exploiting the spectral clustering framework, in order to determine an optimal number of clusters, before the final clustering through a HAC-based approach.

3 x-vector/PLDA Approach

Over the recent years, neural embeddings have been successfully used for speaker recognition and verification [14, 15], outperforming previously used speaker modelling approaches. A feed-forward network trained to separate same-speaker from different-speaker pairs of speech segments is presented in [5]. The network is used in text-independent scenarios – which is a crucial requirement for speaker diarization – and the resulting embeddings are called x-vectors. The same embeddings are used in [6] for the task of speaker diarization, with the proposed architecture learning at the same time not only the audio embeddings, but also the required scoring function. In this work, we are using the x-vectors as our fixed-dimensional embeddings representing the speech segments, but the scoring is done in the PLDA framework, following the baseline approach presented in [16].

PLDA [17, 18] is a generalization of Linear Discriminant Analysis (LDA), first proposed to tackle computer vision problems. It provides a framework in which each data point is considered to be the output of a model which incorporates both within-individual and between-individual variation. In the language of speech processing and speaker embeddings, each x-vector $\mathbf{v}_i$ is assumed to be decomposed as [19]

$$\mathbf{v}_i = \mathbf{m} + \mathbf{\Phi}\boldsymbol{\beta}_i + \mathbf{\Gamma}\boldsymbol{\alpha}_i + \mathbf{e}_i, \quad \boldsymbol{\beta}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad \boldsymbol{\alpha}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad \mathbf{e}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$$

where $\mathbf{m}$ is a global offset, $\mathbf{\Phi}$ is a matirx whose columns form a basis for the speaker-specific subspace, and $\mathbf{\Gamma}$ is a matrix whose columns form a basis for the channel-specific subspace. In the original model $\mathbf{\Sigma}$ is supposed to be diagonal, but in [20], a full matrix $\mathbf{\Sigma}$ is proposed to be used, removing the channel-specific information. Thus, the generative model finally used is

$$\mathbf{v}_i = \mathbf{m} + \mathbf{\Phi}\boldsymbol{\beta}_i + \mathbf{e}_i, \quad \boldsymbol{\beta}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad \mathbf{e}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$$

The parameters $\mathbf{m}, \mathbf{\Phi}, \mathbf{\Sigma}$ are estimated through an Expectation-Maximization (EM) algorithm.

In this framework, the similarity score between two x-vectors $\mathbf{v}_i$ and $\mathbf{v}_j$ (which is the i,j-entry of the affinity matrix $\mathbf{W}$) can be computed via a hypothesis testing. The two hypotheses are a) that $\mathbf{v}_i$ and $\mathbf{v}_j$ have been generated by the same speaker, thus share the same variable $\boldsymbol{\beta}_i = \boldsymbol{\beta}_j$, and b) that $\mathbf{v}_i$ and $\mathbf{v}_j$ have been generated by different speakers, thus $\boldsymbol{\beta}_i \neq \boldsymbol{\beta}_j$. So

$$\begin{aligned}
\mathbf{W}_{ij} &= \log \frac{p(\mathbf{v}_i, \mathbf{v}_j | \text{same speakers})}{p(\mathbf{v}_i | \text{different speakers}) p(\mathbf{v}_j | \text{different speakers})} \\
&= \log \mathcal{N}\left(\begin{bmatrix} \mathbf{v}_i \\ \mathbf{v}_j \end{bmatrix}; \begin{bmatrix} \mathbf{m} \\ \mathbf{m} \end{bmatrix}, \begin{bmatrix} \mathbf{\Phi}\mathbf{\Phi}^T + \mathbf{\Sigma} & \mathbf{\Phi}\mathbf{\Phi}^T \\ \mathbf{\Phi}\mathbf{\Phi}^T & \mathbf{\Phi}\mathbf{\Phi}^T + \mathbf{\Sigma} \end{bmatrix} \right) \\
&\quad - \log \mathcal{N}\left(\begin{bmatrix} \mathbf{v}_i \\ \mathbf{v}_j \end{bmatrix}; \begin{bmatrix} \mathbf{m} \\ \mathbf{m} \end{bmatrix}, \begin{bmatrix} \mathbf{\Phi}\mathbf{\Phi}^T + \mathbf{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}\mathbf{\Phi}^T + \mathbf{\Sigma} \end{bmatrix} \right)
\end{aligned}$$

In order to derive this closed-form formula, we assume that the residual terms $\mathbf{e}_i$, $\mathbf{e}_j$ are independent for all $i \neq j$ and that the latent variables $\boldsymbol{\beta}_i$, $\boldsymbol{\beta}_j$ are also independent when the speakers corresponding to the x-vectors $\mathbf{v}_i$, $\mathbf{v}_j$ are different.

Having established the necessary key notions, the entire system for speaker diarization is summarized as follows:

1. apply any SAD algorithm
2. uniformly partition every speech segment into overlapping short subsegments
3. extract x-vectors for each subsegment
4. compute the PLDA-score between each pair of subsegments $\Rightarrow$ affinity matrix $\mathbf{W}$
5. apply Hierarchical Agglomerative Clustering (HAC) on $\mathbf{W}$ with average linking

4 Incorporating Spectral Clustering

Since we have constructed the affinity matrix $\mathbf{W}$, here we wish to perform the final clustering in the spectral subspace instead of directly applying the HAC. However, in order for the relevant theorems of spectral clustering as presented in [7] to hold, the entries of $\mathbf{W}$ are supposed to be non-negative. Otherwise, it may even be possible that $\mathbf{D}^{-1/2}$ does not exist. In our case, since the entries of $\mathbf{W}$ are the result of a log-likelihood ratio, they may be either positive or negative. In fact, by the way it was defined, if $\mathbf{W}_{ij} > 0$, it means that the same-speaker hypothesis is stronger than the different-speaker one for the x-vectors $\mathbf{v}_i$, $\mathbf{v}_j$, and vice-versa.

In order to overcome this problem, two different approaches are taken. The simplest one is to just shift the entire matrix $\mathbf{W}$ so that all the entries are non-negative. In other words, we find the minimum entry of the matrix and we add its absolute value to every entry. The other approach is based on a generalization of spectral clustering, applied to signed affinity matrices [21]. Following that apporach, we can use the same steps of spectral clustering, with the only difference being that in order to compute the degrees d_i, we sum over the absolute values of the rows of $\mathbf{W}$, so that the diagonal matrix $\mathbf{D}$ has only non-negative entries. The resulting Laplacian matrix after that modification is called the signed Laplacian.

The entire methodolody, following both approaches, is depicted in Algorithm 1.

Algorithm 1 Spectral Clustering for Speaker Diarization.

construct $\mathbf{W}$ following the x-vector/PLDA paradigm $\Rightarrow$ signed $\mathbf{W}$
if signed Laplacian **then**
  $\bar{\mathbf{D}} = \text{diag}\{\bar{d}_1, \bar{d}_2, \cdots\}, \;\; \bar{d}_i = \sum_j |\mathbf{W}_{ij}|$
  $\mathbf{L} = \bar{\mathbf{D}}^{-1/2}\mathbf{W}\bar{\mathbf{D}}^{-1/2}$
else
  $\bar{\mathbf{W}}_{ij} = \mathbf{W}_{ij} + |\min_{ij} \mathbf{W}_{ij}|$
  $\mathbf{D} = \text{diag}\{d_1, d_2, \cdots\}, \;\; d_i = \sum_j \bar{\mathbf{W}}_{ij}$
  $\mathbf{L} = \mathbf{D}^{-1/2}\bar{\mathbf{W}}\mathbf{D}^{-1/2}$
end if
$\mathbf{X} = [\mathbf{x}_1|\mathbf{x}_2|\cdots|\mathbf{x}_k]$; the k largest eigenvectors of $\mathbf{L}$
$\mathbf{Y}_{ij} \triangleq \mathbf{X}_{ij}/\sqrt{\sum_j \mathbf{X}_{ij}^2}$
cluster the rows of $\mathbf{Y}$

5 Experiments & Results

The system proposed is evaluated on the CALLHOME corpus from the National Institute of Standards and Technology (NIST) 2000 Speaker Recognition Evaluation (SRE) Challenge[2]. This is part of a series of challenges that NIST has released over the last years to promote research in the fields of speaker recognition and diarization. The corpus comprises 500 sessions with telephone speech. Each session has a duration ranging from 46sec to 607sec, with the total duration being equal to 17.28h. It is a multilingual dataset (featuring English, Spanish, Japanese, Arabic, Mandarin, and German), while the number of speakers per session varies from 2 to 7. The sampling rate of the audio is 8kHz.

In order to build the sytem, we use the Kaldi speech recognition toolkit [22], and specifically we follow the CALLHOME diarization recipe[3]. The windows to extract the subsegemnts of each speech region have a length of 1.5sec and 50% overlap (subsegment shift = 0.75sec). For each subsegment we compute a sequence of MFCCs, which are the input to the network used to extract the x-vectors. In particular, we use 23-dimensional MFCCs extracted every 10msec from a 25msec-long window.

[2] https://catalog.ldc.upenn.edu/LDC2001S97
[3] https://github.com/kaldi-asr/kaldi/tree/master/egs/callhome_diarization/v2

The DNN architecture is similar to the one described in [23]. Each 1.5sec-long subsegment is initially represented by a sequence of 1.5sec/10msec=150 23-dimensional MFCCs (1 MFCC vector for each frame). The first hidden layer of the DNN sees the current frame, spliced with its neihboring frames in a context window of 2 frames, resulting in a $23 + 2 \times 23 + 2 \times 23 = 115$-dimensional input, with a 512-dimensional output. The next one does a further temporal pooling by splicing the current frame t (which is now the output of the first layer) with the frames $t-2$ and $t+2$ (input dimension = $512 \times 3 = 1536$, output dimension = 512), while the third layer splices t with $t-3$ and $t+3$ (input dimension = 1536, output dimension = 512). No further splicing is done in layers 4 (input dimension = output dimension = 512) and 5 (input dimension = 512, output dimension = 1500). All the first five hidden layers are composed of Rectified Linear Units (ReLUs) with batch normalization. The next layer is called a "statistics pooling" layer. What it does is collect the outputs of the previous layer for all the 150 frames of the subsegment (input dimension = 150×1500) and compute the mean and standard deviation vectors (output dimension = 1500 [for the mean] + 1500 [for the standard deviation] = 3000). The final hidden layer is another ReLU layer with batch normalization (input dimension = 3000, output dimension = 128) and the output layer is a softmax indicating the speaker identity. The embeddings (x-vectors) used are the outputs of the final hidden layer. Thus, finally, each 1.5sec-long subsegment is represented by a 128-dimensional x-vector.

The network is trained using the audio sessions from Switchboard-2[4], Switchboard Cellural[5], and the NIST SREs of 2004[6], 2005[7], 2006[8], and 2008[9]. The training dataset is also augmented with reverberation, noise, and music. The PLDA parameters are estimated on the SRE training subset (not the Switchboard). The x-vectors are further whitened (in order to have identity covariance matrix) and length-normalized [19]. The whitening transformation is computed on in-domain data. To do so, we partition the evaluation dataset into two sets of 250 sessions each, S_1 and S_2. For $i \neq j$, to whiten the x-vectors in S_i, we treat S_j as a held-out set and we estimate the whitening transformation on the x-vectors of S_j.

Finally, for the evaluation, we assume that the oracle SAD information, as well as the number of speakers (clusters) per session are known a priori.

The evaluation metric traditionally used for the task in hand is the Diarization Error Rate (DER), computed as

$$\text{DER} = \frac{\text{False Alarm Speech + Missed Speech + Speaker Error}}{\text{Total Reference Speech}}$$

where the denominator is the duration of speech given the groundtruth labels. As we can see, this metric takes into consideration any potential errors from the SAD. In our case, since we assume that the oracle SAD output is given, the actual error that we compute is given as

$$\text{DER} = \frac{\text{Speaker Error}}{\text{Total Reference Speech}}$$

[4] https://catalog.ldc.upenn.edu/{LDC98S75, LDC99S79, LDC2002S06}
[5] https://catalog.ldc.upenn.edu/{LDC2001S13, LDC2004S07}
[6] https://catalog.ldc.upenn.edu/LDC2006S44
[7] https://catalog.ldc.upenn.edu/{LDC2011S01, LDC2011S04}
[8] https://catalog.ldc.upenn.edu/{LDC2011S09, LDC2011S10, LDC2012S01}
[9] https://catalog.ldc.upenn.edu/{LDC2011S05, LDC2011S08}

When computing the DER, we allow errors within 250msec of a speaker change point. Additionally, overlapping segments are ignored.

The results are given in Table 1. The last column gives the results of the baseline system, without applying spectral clustering, while in the first column we have applied the spectral clustering transformations and the final clustering is done using k-means, as tradditionally done. Following this approach, a significant performance drop is observed. In order to bring closer the two worlds, we also applied spectral clustering with the final clustering of the rows of $\mathbf{Y}$ done via HAC instead of k-means, with the distances computed either in the l_2 or the l_1 space (columns 2 and 3 in Table 1). In any case, the similarity between two clusters, required for the HAC algorithm to be applied, is computed as the average of the pairwise similarities between the x-vectors in the clusters (average linking).

	spectral k-means	spectral HAC (l_2)	spectral HAC (l_1)	baseline HAC (PLDA)
signed	17.05	11.20	11.52	**6.96**
unsigned	15.69	11.64	9.47	

Table 1: DER(%) using the baseline system, or the proposed approach with signed or unsigned Laplacian and with the final clustering step done either with k-means or with HAC.

When the final clustering is done through the bottom-up iterative approach (HAC), the results get significantly improved. Additionally, using the shifted version of $\mathbf{W}$ seems to be beneficial when compared to the use of the signed Laplacian. Overall, the best performance using spectral clustering is achieved through the HAC approach in the l_1 space with the unsigned Laplacian. However, even that design cannot beat the baseline system. This is in accordance with previously published results [11], where spectral clustering was used in the i-vector/cosine distance framework when the number of clusters was known and was compared with a simple k-means algorithm.

6 Conclusion

In this work we tried to incorporate spectral clustering in the x-vector/PLDA framework for speaker diarization, which is a state-of-the-art approach for the problem. We proposed two ways to tackle the "problem" of the signed affinity matrix, with a simple global shift giving the best performance, while the final clustering was suggested to be done in an agglomerative way, as is the standard method in diarization. However, our method was not able to beat the baseline system where the clustering is done directly on x-vectors without any spectral transformation. In the future, we plan to try some additional refinements of the spectral transformation and the affinity matrix, in order to analyze their effect on the final performance.

References

[1] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Transactions on audio, speech, and language processing*, vol. 14, no. 5, pp. 1557–1565, 2006.

[2] 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.

[3] S. Shum, N. Dehak, E. Chuangsuwanich, D. Reynolds, and J. Glass, "Exploiting intra-conversation variability for speaker diarization," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

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

[5] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*, pp. 165–170, IEEE, 2016.

[6] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 4930–4934, IEEE, 2017.

[7] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in neural information processing systems*, pp. 849–856, 2002.

[8] D. P. Ellis and J. C. Liu, "Speaker turn segmentation based on between-channel differences," 2004.

[9] H. Ning, M. Liu, H. Tang, and T. S. Huang, "A spectral clustering approach to speaker diarization," in *Ninth International Conference on Spoken Language Processing*, 2006.

[10] N. Bassiou, V. Moschou, and C. Kotropoulos, "Speaker diarization exploiting the eigengap criterion and cluster ensembles," *IEEE transactions on audio, speech, and language processing*, vol. 18, no. 8, pp. 2134–2144, 2010.

[11] S. Shum, N. Dehak, and J. Glass, "On the use of spectral and iterative methods for speaker diarization," in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[12] J. Luque and J. Hernando, "On the use of agglomerative and spectral clustering in speaker diarization of meetings," in *Odyssey 2012-The Speaker and Language Recognition Workshop*, 2012.

[13] T. H. Nguyen, H. Li, and E. S. Chng, "Cluster criterion functions in spectral subspace and their application in speaker clustering," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 4085–4088, IEEE, 2009.

[14] E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification.," in *ICASSP*, vol. 14, pp. 4052–4056, Citeseer, 2014.

[15] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 5115–5119, IEEE, 2016.

[16] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe, *et al.*, "Diarization is hard: Some experiences and lessons learned for the jhu team in the inaugural dihard challenge," *Proc. Interspeech 2018*, pp. 2808–2812, 2018.

[17] S. Ioffe, "Probabilistic linear discriminant analysis," in *European Conference on Computer Vision*, pp. 531–542, Springer, 2006.

[18] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 1–8, IEEE, 2007.

[19] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[20] P. Kenny, "Bayesian speaker verification with heavy-tailed priors.," in *Odyssey*, p. 14, 2010.

[21] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. W. De Luca, and S. Albayrak, "Spectral analysis of signed graphs for clustering, prediction and visualization," in *Proceedings of the 2010 SIAM International Conference on Data Mining*, pp. 559–570, SIAM, 2010.

[22] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584, IEEE Signal Processing Society, 2011.

[23] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," *Submitted to ICASSP*, 2018.