*INTERSPEECH 2020*
October 25–29, 2020, Shanghai, China

# Speaker Diarization System based on DPCA Algorithm For Fearless Steps Challenge Phase-2

*Xueshuai Zhang*[1,2], *Wenchao Wang*[1,2], *Pengyuan Zhang*[1,2]

[1]Key Laboratory of Speech Acoustics and Content Understanding, Institute of Acoustics, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences

zhangxueshuai@hccl.ioa.ac.cn, wangwenchao@hccl.ioa.ac.cn, zhangpengyuan@hccl.ioa.ac.cn

## Abstract

This paper describes the ASRGroup team speaker diarization systems submitted to the TRACK 2 of the Fearless Steps Challenge Phase-2. In this system, the similarity matrix among all segments of an audio recording was measured by Sequential Bidirectional Long Short-term Memory Networks (Bi-LSTM), and a clustering scheme based on Density Peak Cluster Algorithm (DPCA) was proposed to clustering the segments. The system was compared with the Kaldi Toolkit diarization system (x-vector based on TDNN with PLDA scoring model) and the Spectral system (similarity based on Bi-LSTM with Spectral clustering algorithm). Experiments show that our system is significantly outperforms above systems and achieves a Diarization Error Rate (DER) of 42.75% and 39.52% respectively on the Dev dataset and Eval dataset of TRACK 2 (Fearless Steps Challenge Phase-2). Compared with the baseline Kaldi Toolkit diarization system and Spectral Clustering algorithm with Bi-LSTM similarity models, the DER of our system is absolutely reduced 4.64%, 1.84% and 8.85%, 7.57% respectively on the two datasets.

**Index Terms**: speaker diarization, speaker cluster, Density Peak Clustering algorithm

## 1. Introduction

Speaker diarization is the task of identifying *who spoke when?* in a multi-talker speech recording. It is an important module for a wide variety of applications such as information retrieval from broadcast news, rich transcription for automatic speech recognition (ASR) systems. But diarization is challenging when the speech recording contains unknown number of speakers with variable speech duration, short conversational turns, overlapped speech, noise and reverberation [1, 2, 3].

Most common systems have following five components:(i) a voice activity detector (VAD) [4] is used to removes non-speech regions from the speech recording [5, 6]; (ii) speaker change detection/segmentation [7, 8]; (iii) speaker embedding extraction; (iv) speaker clustering; (v) resegmentation.

For the embedding extraction module, recent work has shown that the diarization performance can be significantly improved by replacing i-vectors [9, 10, 11] with neutral network embeddings, such as d-vectors [12, 13], or x-vectors [14, 15]. So, in our system, we employed x-vectors as our speaker embeddings. In speaker diarization system, speaker clustering module is another important module. There are many clustering algorithms have been employed for diarization system, such as bottom-up agglomerative hierarchical clustering (AHC) [16], Cosine K-means [17], top-down approach [18], PLDA i-vector scoring [19], Spectral clustering coupled with Bi-LSTM similarity models [20, 21]. In our system, we proposed Density Peak Clustering Algorithm (DPCA) as our clustering module.

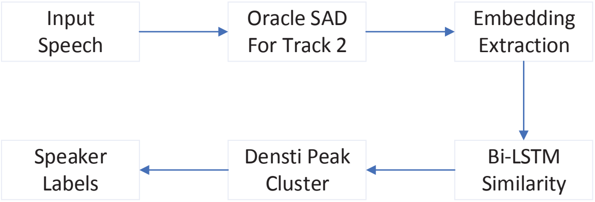

Figure 1: *The data flow of our proposed diarization system.*

DPCA was proposed by Alex Rodriguez and Alessandro laio in Science 2014 [22]. The main idea of this algorithm is to find the high density region which is separated by the low density region. Similarly, DPCA is also based on the assumption that: (i) the local density of the center point of the cluster is higher than that of the neighbor point; (ii) The distance between the center point of cluster with the higher density point is relatively large. The algorithm spends most of its execution time on calculating the local density and the separation distance for each data point in the dataset. This nonparametric algorithm is applicable to any shape of dataset, especially non-spherical datasets, and the algorithm result is not sensitive to parameter selection. DPCA is an unsupervised clustering method, which is suitable for the scenario of unknown number of speakers. But the selection of clustering center points needs to be taken manually, which limits its application in large-scale data sets. Our system propose a modified DPCA method to select clustering centers automatically. First, we use x-vectors of all segments to measure the similarity matrix among all segments. Then, we translate the similarity matrix into a distance matrix. Finally, the proposed DPCA method is applied on the distance matrix to further improve the performance. Since the second Fearless Steps Challenge Phase-2 provides a development dataset in diverse and challenging acoustic conditions [23, 24], we use this dataset to test the performance of our system, Experimental results show that the diarization error rate (DER) is significantly decreased by the proposed method.

## 2. Speaker Diarization System

In our system, an oracle VAD is employed to remove non-speech regions in audios. This section provides a description of the main steps of our system. The over data flow of our diarization system is shown in Figure 1. we will describe all the methods we have experimented with and compare the performance with various combinations of modules.

http://dx.doi.org/10.21437/Interspeech.2020-1666

### 2.1. Feature Extraction

x-vector is proposed in paper [25], which is used to describe the embedding features extracted from time-delay neural network (TDNN). First, 23 dimensional MFCC are extracted from raw audio signal (8000 Hz) with 25ms frame-length and 15ms overlap. Second, MFCCs are fed into a TDNN for supervised learning. Finally, In the TDNN architecture, speaker embeddings are extracted from the affine layer on top of the statistics pooling layer of the classifier network.

### 2.2. Probabilistic Linear Discriminant Analysis

Probabilistic linear discriminant analysis (PLDA) [26, 27] has been successfully used in speaker recognition to measure the similarity between two speakers. It assumes that each speaker embedding $\boldsymbol{x}_i$ is modeled as:

$$\begin{aligned} \boldsymbol{x_i} &= \boldsymbol{m_u} + \boldsymbol{V}\boldsymbol{y_i} + \boldsymbol{\xi_i} \\ \boldsymbol{\xi} &\sim N(\boldsymbol{0}, \boldsymbol{I}) \\ \boldsymbol{\xi} &\sim N((\boldsymbol{0}, \boldsymbol{\Sigma}) \end{aligned} \tag{1}$$

where $\boldsymbol{m}_u$ is the means of embeddings, $\boldsymbol{V}$ is the matrix which referred to as eigenvoices, $\boldsymbol{y}_i$ is the speaker dependent latent variable with a Normal distribution $N(\boldsymbol{0}, \boldsymbol{I})$, $\boldsymbol{\xi_i}$ denotes the normally distributed residual noise with zero mean and full covariance matrix. Expectation-maximization (EM) algorithm is used to estimate the parameters of the PLDA model [26, 27]

### 2.3. Similarity Measurement with LSTM

Speaker similarity measurement with LSTM was proposed by Lin [20]. In a similarity matrix $\boldsymbol{S}$, the values are means whether each segment pair is from the same speaker. Each row of the matrix was predicted by a stacked Bi-LSTM using the binary cross entropy (BCE) loss function. x-vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are catenated as the 2d-dimensional input [x$_i^T$, x$_j^T$]$^T$, where the corresponding output is $S_{ij}$. The i$^{th}$ row of similarity matrix $\boldsymbol{S}$ can be described as follows:

$$\boldsymbol{S}_i = [S_{i1}, S_{i2}, ...S_{in}] = f_{lstm}\left(\begin{bmatrix} x_i^T \\ x_1^T \end{bmatrix}, \begin{bmatrix} x_i^T \\ x_2^T \end{bmatrix} ... \begin{bmatrix} x_i^T \\ x_n^T \end{bmatrix}\right) \tag{2}$$

where $\boldsymbol{x}_i$ is the speaker embedding of i$^{th}$ audio segment. The architecture of the neural networks includes two Bi-LSTM layers, followed by two fully connected layers. Both of the Bi-LSTM layers have 512 outputs (256 forward and 256 backward). The first fully connected layer is 64-dimensional with the ReLU activation function. The second fully connected layer is 1-dimentional with a sigmoid function to output a similarity score.

### 2.4. AHC-clustering

AHC clustering is a method of cluster analysis which seeks to build a hierarchy of clusters [16], Strategies for hierarchical clustering can be described as following steps:

1. Each segment is initialized as a single cluster, and calculate the minimum distance between all clusters.
2. Merge the two classes with the smallest distance into a new class, and update distances of clusters to the new cluster.
3. Repeat the previous step until the specified number of clusters has been obtained.

### 2.5. Spectral Clustering

In Spectral clustering (SC) [20, 21], the segments are treated as nodes of a graph. Then, clustering is treated as a graph partioning problem. The nodes are mapped into a low-dimensional space which can be easily segregated to form clusters. Strategies for SC can be described as below:

1. Input Matrix: Given similarity matrix $\boldsymbol{S}$ which each element is the similarity between two segments and set diagonal elements to 0.
2. Symmetrization: $Y_{i,j}$ = max($S_{ij}$, $S_{ji}$)
3. Diffusion: $Y \leftarrow YY^T$
4. Normalization: $S_{ij}$ = $Y_{ij}$/max$_k Y_{ik}$
5. Get Laplacian matrix by the following formula:

$$\begin{aligned} d_i &= \sum_{n=1}^{N} a_{ik} \\ \boldsymbol{D_c} &= diag\{d_1, d_2, ..., d_N\} \\ \boldsymbol{L_c} &= \boldsymbol{D_c} - \boldsymbol{S} \end{aligned} \tag{3}$$

6. Perform Singular Value Decomposition (SVD): Compute eigenvalues and eigenvectors of $\boldsymbol{L_c}$
7. Create an eigen gap vector and find the argument max of the vector $\mathbf{e}_c$

$$\begin{aligned} \boldsymbol{e_c} &= [\lambda_2 - \lambda_1, \lambda_3 - \lambda_2, ....\lambda_N - \lambda_{N-1}] \\ ns &= min(argmax(\boldsymbol{e_c}), N_s) \end{aligned} \tag{4}$$

where we can cap the maximum speaker number $N_s$.

8. Take the $n_s$ smallest eigenvalues $\lambda_1$,$\lambda_2$,...$\lambda_{ns}$ and corresponding eigenvectors. For all segments, these eigenvectors are considered as *ns* dimensional Spectral embeddings. Spectral embeddings are then clustered by K-means clustering algorithm.

### 2.6. Proposed Density Peak Clustering algorithm (DPCA)

Density Peak Clustering algorithm [22] is a new density-based clustering method. The advantage of this method is that it can detect non-convex clusters and outliers. The main idea of this method is to find the high density regions which was separated by low density regions. The algorithm is based on these assumptions: (1) The density of the center of a cluster is higher than its neighbors; (2) The distance between the center point of cluster and the higher density point is relatively large.

$$\begin{aligned} \rho_i &= \sum_{x_j \in U} \chi(dist(x_i, x_j) - dc) \\ \chi(x) &= \begin{cases} 1 & x \leq 0 \\ 0 & x > 0 \end{cases} \end{aligned} \tag{5}$$

where *dc* is a cutoff distance, and often set as 2% of the data size [22]. $\rho_i$ is equal to the number of points that are closer than *dc* to point *i*. The algorithm is sensitive only to the relative magnitude of $\rho_i$, which means that, for large data sets, the result of the analysis is robust with respect to the choice of *dc*.

$$\theta_i = \begin{cases} min(d_{ij}) & \rho_j > \rho_i \\ max(d_{ij}) & \rho_j \leq \rho_i \end{cases} \tag{6}$$

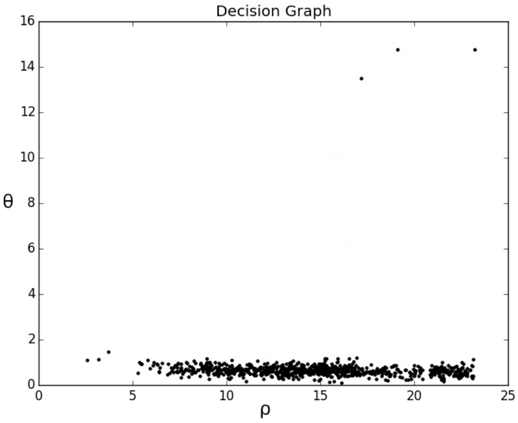

Figure 2: *Decision Graph.*

where $\theta_i$ is the cluster center distance. The formula (6) denotes that: (1) when the density of the point is not the largest of all points, then the point is not the center point, the distance will be set as the distance between the point and its nearest point; (2) when the density of the point is the largest of all points, then the point is the center point, the distance will be set as the distance between the point and its farthest point. when $\rho$ and $\theta$ of each data are calculated, each point $x_i$ is depicted on a decision graph by using ($\rho[i]$, $\theta[i]$) as its x-y coordinate as shown in Figure 2. By observing the decision graph, the cluster centers can be identified in the top region since they are with relatively large $\rho[i]$ and large $\theta[i]$ [22].

**Algorithm 1** Transforming similarity matrix into distance matrix

```
Require:
  The similarity matrix between all segment, S[n][n]

Ensure:
  The distance matrix between all segment, DS[n][n]
  for $i \leftarrow$ 1 to n do
      for $j \leftarrow$ i to n do
          S[i][j] = max(S[i][j], S[j][i]);
          S[j][i] = max(S[i][j], S[j][i]);
      end for
  end for
  for $i \leftarrow$ 1 to n do
      for $j \leftarrow$ i to n do
          if i == j then
              sv = S[i][j];
              continue;
          end if
          DS[i][j] = S[i][j] * -1 + sv;
      end for
  end for
```

In our system, first, we translate the similarity matrix ***S*** to the distance matrix ***DS*** by using Algorithm 1 and calculate the $\boldsymbol{\rho}$ and $\boldsymbol{\theta}$ of each segment. Second, when the $\boldsymbol{\rho}$ and $\boldsymbol{\theta}$ are calculated, we set $\gamma_i = \rho_i * \theta_i$ and sort the $\boldsymbol{\gamma}$ in a descending order, then find the postion index k of the largest gap in the descending queue. The index k is the cluster number, the points before index k in the descending order are cluster centers. Finally, we use the DPCA algorithm on top of the ***DS*** to clustering segments goes as Algorithm 2.

**Algorithm 2** DPCA Clustering algorithm

```
Require:
  Distance matrix, S[n][n]
  cut-off distance, dc
  The maximum number of cluster centers, $n_s$
Ensure:
  The local density, $\rho[]$
  The dependent cluster center distance, $\theta[]$
  The point-cluster assignment cluster, label[]
  The cluster number of the data, $n_{spk}$

  for $i \leftarrow$ 1 to n do
      $\rho[i] \leftarrow$ calculate $\rho[i]$ by equation (5) based on S and dc;
  end for
  $\rho$ = SortDecend($\rho$) //sort the $\rho$ in a descending order;
  $\theta[] \leftarrow$ calculate each $\theta$ by equation (6) based on $\rho$

  for $i \leftarrow$ 1 to n do
      $\gamma[i] = \rho[i] * \theta[i]$;
  end for
  $\gamma$ = SortDecend($\gamma$) //sort the $\gamma$ in a descending order;
  $\gamma_{sub}$ = maximum($\gamma$,$n_s$) //obtaining previous $n_s$ values of $\gamma$;
  for $i \leftarrow$ 1 to $n_s$ do
      k[i] = $\gamma_{sub}[i+1]$ / $\gamma_{sub}[i]$;
  end for
  $n_{spk}$ = Arg(Max(k[i])); // index of maximum value
  for $i \leftarrow$ 1 to $n_{spk}$ do
      //obtainning the cluster centers
      Centers[] $\leftarrow$ The index of data point corresponding $\gamma[i]$
  end for
  //assign points to each cluster based on $\rho[]$, $\theta[]$ and Centers[]
  cluster[] $\leftarrow$ assign points to clusters
```

# 3. Experimental Results

In this section, we will compare our proposed system with two baseline systems in the TRACK 2 dataset of Fearless Steps Challenge Phase-2 [23, 24]. The experiment results show that the performance of our proposed system is better than the other systems.

## 3.1. Data

In our proposed system, models for extracting x-vectors are trained on a collection of SRE-database including SRE 2004, 2005, 2006, 2008 and Switchboard. All recordings are sampled at 8 kHz. Augmented data was generated by Kaldi Toolkit [28] using the MUSAN and RIR datasets[1]. The reason for this process is to simulate the distortions typical to far-field microphone under noisy environments. Reverberation was also performed using the impulse response generator based on [29].

The TRACK 2 dataset of Fearless Steps Challenge Phase-2 includes three datasets: Train dataset, Dev dataset and Eval dataset. The Train dataset consists of 125 audio streams each of length 30 minutes with the number of speakers ranging from 4 to 61. The Dev dataset consists of 30 audio streams each of length 30 minutes with the number of speakers ranging from 7

[1] http://www.openslr.org

Table 1: *TRACK 2: Reference SAD, FEARLESS STEPS CHALLENGE Dataset*

| **Systems** | **(0)** | **(1)** | **(2)** | **(3)** |
|---|---|---|---|---|
| Embedding Extraction | — | x-vector | x-vector | x-vector |
| Distance Measure | — | PLDA | Bi-LSTM | Bi-LSTM |
| Clustering Algorithm | — | AHC | Spectral | DPCA |
| DER:Dev Set | 68.68% | 47.39% | 44.59% | **42.75%** |
| DER:Eval Se | 67.91% | 48.37% | 47.09% | **39.52%** |

to 61. All these datasets are comprised of three mission critical stages from the NASA's Apollo-11 mission, and most of the audios are suffer from a wide range of issues like high channel noise, system noise, attenuated signal bandwidth, transmission noise, cosmic noise, etc. In our system, the Bi-LSTM model and PLDA were trained on the Train dataset. The Dev dataset and Eval dataset were used to test the performance of our system.

### 3.2. Performance metric

The system performance was evaluated in terms of Diarization Error Rate(DER), as defined by NIST [30]. DER includes three components: false alarm (FA), missed detection (Miss), and speaker confusion, The FA and Miss are caused by Speech Activity Detection (SAD). It is common not to evaluate short collars centered on each speech turn boundary (0.25s on both sides). The scoring script of evaluation is provided by the Fearless Steps Challenge organizers [23, 24]. Since the TRACK 2 provides the ground truth labels for SAD for each audio file, we exclude FA and Miss from evaluations.

### 3.3. Baseline system

We compare our system with two baselines. The first baseline is based on x-vector using PLDA similarity measurement and AHC clustering algorithm. The second baseline is based on x-vectors using Bi-LSTM similarity models and Spectral clustering algorithm.

### 3.4. Implementation details

1. **Speech segmentation:** All experiments share the same segmentation. Each audio with ground truth labels for SAD are segmented into sub-segments with length 1.5s and overlap 0.25s.

2. **x-vector extraction:** 23 dimensional MFCCs are extracted from each subsegment and followed by cepstral mean normalization. The whole process of x-vector extraction is described in [25].

### 3.5. Results & Discussions

Table 1 represents the DER of our system and two baseline systems on the Dev dataset of TRACK 2. It should be note that the baseline Kaldi Toolkit diarization system (x-vector based on TDNN with PLDA scoring model and AHC clustering) corresponds to system (1) and the baseline of Fearless Steps challenge organizer corresponds to system (0).

In the system (1), the best stop threshold of AHC algorithm was obtained by training on the Train dataset of TRACK 2. As the result shows in Table 1, it achieves DER = 47.39%. The result shows that the AHC clustering algorithm is not working well on the dataset. This is due to the fact that the Dev dataset of TRACK 2 is very diverse and complex, while PLDA is not suitable to measure the similarity between long segments and short segments of different speakers, as it ignores the contextual information of segments.

As Bi-LSTM takes full advantage from forward and backward sequences, we introduce it into system (2) and using Spectral clustering algorithm to cluster segments, as it can judge the number of speakers automatically. The result in Table 1 shows that the system (2) achieves DER = 44.59%, with DER 2.8% absolute decreases than system (1) on the Dev dataset. This is due to the fact that with sufficient training data, the similarity between two segments which measured by Bi-LSTM is robust against varing scenarios.

But Spectral clustering is more suitable for the problem of balanced classification, which means that the number of segments among clusters are with little differences. So, the Spectral clustering algorithm is not suitable for the TRACK 2 dataset, while the speech duration of each speaker varies greatly in the audio. As DPCA algorithm is applicable to any shape of dataset, especially non-spherical datasets, and the algorithm result is not sensitive to parameter selection, we introduce it into our system and propose a simple method to select clustering centers automatically. The best threshold of *dc* in our system was obtained by training on the Train dataset of TRACK 2. The result in Table 1 shows that the system (3) achieves DER = 42.75%, with DER 1.84% absolute decreases than system (2) on the Dev dataset. By comparing the cluster results of system (1), system (2), and system (3), we found that some short segments of the speaker in system (3) were correctly classified. The reason that DPCA clustering algorithm is better than Spectral clustering algorithm is that it is able to map data with arbitrary dimension onto a 2-dimentional space, and construct hierarchical relationship for all data points on the new reduction space.

## 4. Conclusions

In this paper, we introduce the Density Peak Clustering Algorithm into our system and present using a simple method to judge the number of speakers automatically. As Density Peak Clustering Algorithm can detect non-convex clusters, and can obtain the clusters in a single step regardless of the shape and dimensionality of the space, The DER of our system achieves 4.64%, 1.84% and 8.85%, 7.57% absolute decreases on the Dev and Eval dataset respectively compared with the Kaldi Toolkit diarization system and the Spectral algorithm with Bi-LSTM similarity models.

## 5. Acknowledgements

This work is partially supported by the National Natural Science Foundation of China (Nos. 11590772,11590774,11590770).

# 6. References

[1] S. E. Tranter and D. A. Reynolds, "An overview of auto-matic speaker diarization systems," *IEEE Transactions on Audio,Speech, and Language Processing*, vol. 14, no. 5, pp. 1557–1565, 2006.

[2] J. H. L. Hansen, J. Alberte, N. Jones, H. Dubey, and A. Sang wan, "Multi-stream audio analysis for knowledge extraction and understanding of small-group interactions in peer-led team learning," *Seventh Annual Conference Peer-LedTeam Learning Inter-national Society, the University of Texas at Dallas, Richardson, TX, USA*, pp. 1-1, 2018.

[3] J. H. L. Hansen, H. Dubey, and A. Sangwan, "CRSS-LDNN Long-duration naturalistic noise corpus containing multi-layer noise recordings for robust speech processing," *The Journal of the Acoustical Society of America*, vol. 144, no. 3, pp. 1797–1797, 2018.

[4] M. Price, J. Glass, and A. P. Chandrakasan, "A low-power speech recognizer and voice activity detector using deep neural networks," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 1, pp. 66–75, Jan 2018.

[5] H. Dubey, A. Sangwan, and J. H. L. Hansen, "Leveraging Frequency-Dependent Kernel and DIP-based Clustering for Robust Speech Activity Detection in Naturalistic Audio Streams," *IEEE/ACM Trans. on Audio, Speech and Language Processing*, vol. 26, no. 11, pp. 2056–2071, 2018.

[6] H. Dubey, A. Sangwan and J. H. L. Hansen, "Robust Feature Clustering for Unsupervised Speech Activity Detection," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB*, 2018, pp. 2726-2730.

[7] M. Hrúz and Z. Zajíc, "Convolutional Neural Network for speaker change detection in telephone speaker diarization system," *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA*, 2017, pp. 4945-4949.

[8] R. Yin, H. Bredin, and C. Barras, "Speaker change detection in broadcast tv using bidirectional long short-term memory networks," *in Proc. Interspeech 2017*, 2017, pp. 3827–3831.

[9] S. Cumani, N. Brümmer, L. Burget, P. Laface, O. Plchot, V. Vasilakakis, "Pairwise discriminative speaker verification in the i-vector space," *IEEE Trans. Audio Speech Lang. Process.*, vol. 21, no. 6, pp. 1217-1227, Jun. 2013..

[10] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans.on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[11] M. Senoussaoui, P. Kenny, N. Brummer, P. Dumouchel, "Mixture of PLDA models in i-vector space for gender–independent speaker recognition," *Proc. INTERSPEECH*, pp. 25-28, 2011..

[12] Q. Wang, C. Downey, L. Wan, P. A. Mansfield and I. L. Moreno, "Speaker diarization with LSTM," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB*, 2018, pp. 5239-5243.

[13] L. Wan, Q. Wang, A. Papir and I. L. Moreno, "Generalized End-to-End Loss for Speaker Verification," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB*, 2018, pp. 4879-4883.

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

[15] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey and S. Khudanpur, "X-Vectors: Robust DNN Embeddings for Speaker Recognition," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB*, 2018, pp. 5329-5333.

[16] H. Sun, B. Ma, S. Z. K. Khine and H. Li, "Speaker diarization system for RT07 and RT09 meeting room audio," *2010 IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, TX*, 2010, pp. 4982-4985.

[17] F. Castaldo, D. Colibro, E. Dalmasso, P. Laface and C. Vair, "Stream-based speaker segmentation using speaker factors and eigenvoices," *2008 IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, NV*, 2008, pp. 4133-4136.

[18] S. Meignier, D. Moraru, C. Fredouille, J. F. Bonastre, and L. Besacier, "Step-by-step and integrated approaches in broadcast news speaker diarization," *Computer Speech & Language*, vol. 20, no. 2-3, pp. 303–330, 2006.

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

[20] Lin, Q. Yin, R. Li, M. Bredin, H. Barras, C. L, "LSTM based Similarity Measurement with Spectral Clustering for Speaker Diarization," *Proc. Interspeech*, pp. 366-370, 2019.

[21] S. Shum, N. Dehak, and J. Glass, "On the Use of Spectral and Iterative Methods for Speaker Diarization," *Proc. Interspeech 2012*, pp. 482-485, 2012.

[22] Rodriguez, Alex, and Alessandro Laio, "Clustering by fast search and find of density peaks," *Science*, vol. j344, no. 6191, pp. 1492-1496, 2014.

[23] John H. L. Hansen and Aditya Joglekar and Meena Chandra Shekhar and Vinay Kothapally and Chengzhu Yu and Lakshmish Kaushik and Abhijeet Sangwan, "The 2019 Inaugural Fearless Steps Challenge: A Giant Leap for Naturalistic Audio," *Proc. Interspeech 2019*, 2019, pp. 1851-1855.

[24] John H.L. Hansen and Abhijeet Sangwan and Aditya Joglekar and Ahmet E. Bulut and Lakshmish Kaushik and Chengzhu Yu, "Fearless Steps: Apollo-11 Corpus Advancements for Speech Technologies from Earth to the Moon," *Proc. Interspeech 2018*, 2018, pp. 2758-2762.

[25] D. Snyder, D. Garcia-Romero, D. Povey, S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," *Proc. of Interspeech*, pp. 999-1003, Aug 2017.

[26] S. Ioffe, "Probabilistic linear discriminant analysis," *Proc. Eur. Conf. Comput. Vis.*, pp. 531-542, 2006.

[27] S. J. D. Prince and J. H. Elder, "Probabilistic Linear Discriminant Analysis for Inferences About Identity," *007 IEEE 11th International Conference on Computer Vision*, Rio de Janeiro, 2007, pp. 1-8.

[28] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey and S. Khudanpur, "X-Vectors: Robust DNN Embeddings for Speaker Recognition," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB, 2018, pp. 5329-5333.

[29] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[30] J. G. Fiscus, N. Radde, J. S. Garofolo, A. Le, J. Ajot, and C. Laprun, "The Rich Transcription 2006 Spring Meeting Recognition Evaluation" *Machine Learning for Multimodal Interaction*, vol. 4299, pp. 309–322, 2006.