*INTERSPEECH 2021*
30 August – 3 September, 2021, Brno, Czechia

# Adapting Speaker Embeddings for Speaker Diarisation

*Youngki Kwon, Jee-weon Jung, Hee-Soo Heo, You Jin Kim, Bong-Jin Lee, Joon Son Chung*

Naver Corporation, South Korea
youngki.kwon@navercorp.com

## Abstract

The goal of this paper is to adapt speaker embeddings for solving the problem of speaker diarisation. The quality of speaker embeddings is paramount to the performance of speaker diarisation systems. Despite this, prior works in the field have directly used embeddings designed only to be effective on the speaker verification task. In this paper, we propose three techniques that can be used to better adapt the speaker embeddings for diarisation: dimensionality reduction, attention-based embedding aggregation, and non-speech clustering. A wide range of experiments is performed on various challenging datasets. The results demonstrate that all three techniques contribute positively to the performance of the diarisation system achieving an average relative improvement of 25.07% in terms of diarisation error rate over the baseline.
**Index Terms**: speaker diarisation, feature enhancement, dimensionality reduction, clustering.

## 1. Introduction

Speaker diarisation addresses the task of segmenting multi-speaker audio recordings into homogeneous single speaker segments, effectively solving *"who spoke when"*. The task is a valuable pre-processing step for understanding and transcribing conversations. The ability to determine who spoke when facilitates a rich transcription of conversation providing context as well as the content.

Recent works on speaker diarisation can be divided into two strands. The first strand trains speaker diarisation systems in an end-to-end manner [1]. Specifically, several attempts have been made to directly perform speaker diarisation from audio input by constructing deep neural networks (DNNs) with multiple recurrent layers and training them with permutation invariant training technique [2–4]. However, the existing end-to-end systems only show good performance in constrained settings and do not generalise well to real-world conditions.

The second strand of work utilises the traditional pipeline which typically comprises separate modules for each of the steps that must be performed for speaker diarisation. Although the exact specification differs from one work to another, the majority of works use at least three independent components: a speech activity detection (SAD) module that detects speech segments from an input audio, an embedding extraction module that extracts speaker representations (*i.e.* embeddings), and a clustering module that maps the extracted embeddings to the clusters of unknown numbers [5, 6]. The weakness of the approach is that the individual components, particularly the SAD module and the speaker embedding extractor, are pre-trained and not optimised for the diarisation task.

Despite this, most of the best performing systems in recent challenges are based either on the traditional pipeline or a hybrid of the two types (traditional and end-to-end systems) [7, 8]. Moreover, recent state-of-the-art approaches based on the Bayesian hidden Markov model [9–11] and target speaker voice activity detection [12] also require initial diarisation results to bootstrap their operation, which highlights the importance of the traditional baseline system. The focus of this work will therefore be on complementing the SAD and improving the embeddings for the traditional pipeline.

To this end, we propose two additional steps on top of traditional speaker diarisation systems. First, we propose techniques that enhance the existing embedding to be suitable for the speaker diarisation task. In the existing systems, the embeddings trained for speaker verification are directly used for diarisation. However, there should be differences in the use of features, despite the fact that the embeddings should share similar representations of speakers. For example, the speaker verification task requires discrimination for more than thousands of speakers at a time, whereas the diarisation task requires discrimination only for ten or fewer speakers. There is also a difference in that the duration for comparing speakers is shorter in diarisation compared to speaker verification. Typically, the VoxCeleb datasets [13–15], which are widely used in speaker verification, use speech segments over ten seconds on average in order to compare speakers. On the other hand, in diarisation, the comparison should be performed at shorter intervals, for example between one or two seconds due to the issue of unknown speaker changing points. Considering the differences between these tasks, we propose to reduce the dimensionality of existing embeddings and to enhance the representative power of the speaker within a session, further described in Sections 3.1 and 3.2 respectively.

Second, we propose a technique referred to as the non-speech clustering that complements existing SAD. In existing works, the SAD module is trained and adopted independently of feature extraction and clustering steps. This could cause inefficiency in the overall pipeline, considering that the operations of other modules highly depend on the results from SAD performed first. We train the embedding extractor to represent non-speech segments as well as different speakers, as described in Section 3.3. Not only does this improve the SAD performance by introducing an ensemble-like effect with the existing SAD module, it also helps to reduce speaker confusion error by excluding less confident embeddings in the clustering process.

Experiments are performed on two datasets from the previous DIHARD challenges and an internal dataset of real-world conversations. The proposed methods consistently demonstrate significant improvements over baselines across many different settings, and the performances of our best systems exceed the state-of-the-art in many of the challenge sub-tasks.

## 2. Baseline system

This section describes the baseline diarisation system for the experiments in this paper.

### 2.1. Speech activity detection

Speech activity detection (SAD) detects the onsets and offsets of the continuous speech segments in the input audio session. Our baseline SAD module is identical to [16]. First, SAD is performed on 25 millisecond windows of 40-dimensional mel-filterbank features, shifting 10 millisecond at a time. To utilise short-term context information, we stack five frames from both left and right to form a 440-dimensional input feature. The extracted features are fed into a multi-layer perceptron (MLP) with 3 hidden layers to decide whether each frame contains speech. Each hidden layer in the MLP network contains 512 nodes ac-



http://dx.doi.org/10.21437/Interspeech.2021-448

tivated by the leaky ReLU function. The MLP network is trained using the development sets of DIHARD I [17], DIHARD II [18] and DIHARD III [7], as well as the VoxConverse [19] and AMI [20] datasets.

Based on the SAD results, we split each session into continuous speech segments by compensating for the excessively rapid changes in SAD results following [21]. We decide the onsets and offsets by sliding a window of a certain size (100ms in our configuration). In particular, the onset is identified when the ratio of speech-activated frames exceeds 70% in the window, and the offset is also identified following the same rule for non-speech frames.

### 2.2. Speaker embedding extraction

In this step, we extract the features representing speaker characteristics from the speech segments detected in the previous step. For the embedding extraction, we train the **H / ASP** architecture described in [22] with a few modifications [23–26]. First, we use the development set of both VoxCeleb 1 and 2 [14, 15]. The number of filters in the first convolutional layer is configured to 64. We adopt an average pooling instead of attentive statistics pooling for aggregation. The angular margin softmax [27] objective function is used to train the model.

To extract a speaker embedding from a segment, we first extract a 256-dimensional speaker embedding using a window of 1.5 second width and 0.5 second shift and then apply average to the extracted embeddings.

### 2.3. Clustering

Using the extracted speaker embeddings, we then generate speaker labels for each speech segment using two well-known clustering algorithms; agglomerative hierarchical clustering (AHC) and spectral clustering (SPC) [28, 29]. AHC constructs a hierarchy based on the distances between features and forms a group. For AHC, it is common to manually set a distance threshold to estimate the number of speakers. As an alternative, a silhouette coefficient-based trick can be applied to estimate the number of speakers with the expectation of better generalization performance for various environments [30, 31].

SPC groups the features using the manifold of embedding space [29, 32]. First, the affinity matrix where each element represents cosine similarity between two features is calculated. Then, we apply eigen-decomposition to the affinity matrix without any additional refinement process [33]. A threshold of 20 is empirically applied to the eigen-values for determining the number of clusters. Finally, we perform k-means clustering on the spectral embeddings, which is a set of eigen-vectors corresponding to the largest eigen-values, to estimate the final cluster labels.

## 3. Proposed speaker embedding adaptation

This section introduces three proposed techniques. Figure 1 illustrates modified process pipeline including these techniques. All three techniques can be independently adopted. Thus, certain combination of these techniques can be applied to the existing pipeline.

### 3.1. Dimensionality reduction

Speaker verification and diarisation's required characteristics differ, thus, it would be beneficial to adapt the speaker embeddings for diarisation task. In speaker diarisation, embeddings are used to represent only a small number of speakers in one session, different from those in speaker verification required to represent unlimited number of speakers. We argue that the embeddings extracted for speaker verification might be excessively high-dimensional, thus, sparse for speaker diarisation. For instance, only a small part of the information included in the embeddings would be used to distinguish a small

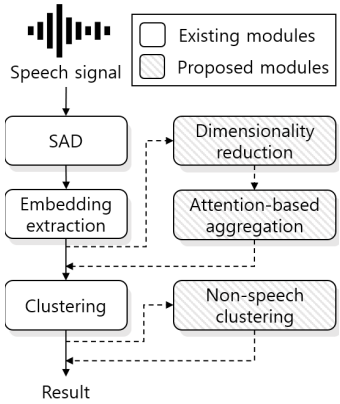


Figure 1: *Process pipeline of common speaker diarisation system with the proposed modules.*

number of speakers, even though the 256-dimensional embeddings are extracted. In addition, the remaining information included in the embeddings would have uncertainty due to factors other than the speaker information and cause noise in the affinity matrix. To mitigate such a problem, we propose the dimensionality reduction method that finds the suitable low-dimensional code for each session. Note that we find the appropriate representation with low dimensionality for each session rather than finding the global representation.

To achieve this goal, we propose to adopt an auto-encoder (AE) with low-dimensional bottleneck (i.e., code). After random initialization, we train the AE to minimise the mean square loss between the original embeddings and the embeddings reconstructed from the network during run-time. The AE comprises two layers, one for the encoder and the other for the decoder. For the encoder layer, we apply max feature-map activation, well-known for training compact representation [34]. Using the trained AE, the original embeddings with 256 dimensions are projected into 20-dimensional space. We train the AE by 200 epochs using adam optimizer with a 0.001 learning rate for each session.

### 3.2. Attention-based embedding aggregation

Despite the success of AHC and SPC in speaker diarisation, these algorithms have inherent limitations depending on the input features. These limitations should be handled at the embedding-level rather than improving the clustering methods. For example, SPC is sensitive to noises in the affinity matrix. Affected by outliers in embedding space, AHC often fails to construct hierarchy and misestimates the number of clusters. To mitigate such problems, we propose a feature enhancement technique for speaker diarisation referred to as attention-based embedding aggregation. The goal of this technique is to remove noises and outliers that may occur on the affinity matrix using the global context within each session.

To do so, we aggregate the features of each cluster by using self-attention, which is common in architectures such as Transformer or its variants [35, 36]. First, we calculate the attention map for each embedding using the softmax function. Subsequently, the embeddings are aggregated based on this attention map. These two steps are repeated several times. Algorithm 1 describes the process.

We expect that the process of aggregating the embeddings for each cluster would remove the noises in the affinity matrix and outliers. In the proposed technique, it is necessary to carefully determine the temperature value, which is applied before the softmax function, so that the proper clusters can be formed by aggregation. With the appropriate value of temperature and the number of repetitions, we

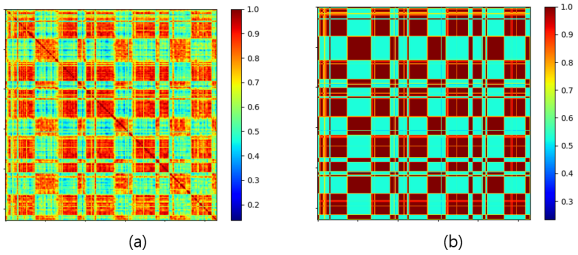


Figure 2: *Effect of the attention-based aggregation technique on the affinity matrix. We calculate the affinity matrix using one sample in development set to compare the before (a) and after (b) applying the attention-based aggregation technique. The results show that the proposed technique act like a refinement process that removes the most of noises on the affinity matrix.*

can perform soft version of clustering with this method. Figure 2 compares the effect of the proposed method on the affinity matrix. Results show that the proposed method effectively removes the noises on affinity matrix and outliers. For more details, two hyper-parameters need to be configured to apply the proposed attention-based aggregation: number of repetitions and temperature value before softmax function. We fix these two values as 5 and 15, respectively.

### 3.3. Non-speech clustering

SAD, the very first building block, has a tremendous impact on speaker diarisation. In most cases, the existing SAD modules are performed independently from embedding extractor or clustering. This design could raise problems, considering the operation of the subsequent steps performed based on the SAD result. For example, if the SAD module detects a speech segment where the embedding extractor hardly represents the speaker characteristics, the embedding would have low reliability and degenerates the result of the clustering.

We design the SAD module that closely interacts with the two subsequent steps. This approach is motivated from [30]. First, we train the embedding extractor to represent both speech and non-speech segments. To do so, we add one additional class that includes non-speech regions when training the speaker embedding extractor. Here, note that the speaker embedding extractor is trained for speaker identification. With this process, the embedding extractor learns how to detect not only the specific speaker but also the absence of the speaker. Therefore, we can handle the non-speech segments as an additional cluster during the clustering step.

We also propose to combine the results from the SAD module and the clustering output to achieve more reliable results. Let the results from the SAD module for each segment be represented as $e_i \in 0,1$ where 0 and 1 represent non-speech and speech label, respectively. Let $c_i \in 0,...,S$ as the clustering results where 0 stands the non-speech cluster and $S$ is the number of speakers in one session.

**Algorithm 1** Attention-based embedding aggregation

**Input:** Speaker embeddings $\mathbf{X}$ of size $L \times 20$
**Hyper-parameters:** Number of repetition $N$, Temperature value $\tau$
**for** $iteration = 1,2,...,N$ **do**
    Construct affinity matrix $\mathbf{A}|\mathbf{A}_{i,j}$=cos($\mathbf{X}_i$,$\mathbf{X}_j$)
    $\mathbf{A}$ = softmax($\mathbf{A}$ *$\tau$)
    $\mathbf{X}$ = dot($\mathbf{A}$, $\mathbf{X}$)
**end for**

The reliable set $R$ can be constructed if $e_i = c_i = 0$, or $e_i = 1$ and $c_i > 0$. Subsequently, the centroid vector $C_j$ for each cluster $j$ is calculated by averaging the embeddings extracted from reliable set $R$. Finally, the refined results can be obtained by mapping the embeddings $E_{1,...,T}$ to the nearest centroid among $C_{0,...,S}$.

The use of more confident embeddings for clustering allows the system to obtain more representative cluster centers, leading to an improvement in speaker confusion error. In addition, in case of track 2, we empirically find an ensemble effect of the external SAD module and the embedding-based non-speech clustering contributes to an improvement in FA and MS metrics.

## 4. Experiments

We conduct experiments to evaluate proposed speaker embedding adaptation techniques on three datasets: the first and the second DIHARD challenge datasets [17, 18] and our internal dataset of real-world conversations recorded with a single-channel microphone. Sections 4.1 and 4.2 describe the evaluation protocol and the baselines, common across all experiments. Section 4.3 describes experiments on each dataset.

### 4.1. Evaluation protocol

We use the diarisation Error Rate (DER) as the primary metric. The DER is the sum of three error components: missed speech (MS, a speaker in reference, but not in prediction), false alarm (FA, a speaker in prediction, but not in reference), and speaker confusion error (SC, assigned to wrong speaker ID) We use the dscore[1] tool to compute the metrics. For the internal dataset, we use a 250ms collar. And for the DIHARD datasets, we use a 0ms collar which is in line with evaluation protocol of the DIHARD challenge.

### 4.2. Baselines

We test the two baselines described in Section 2. The two baselines each utilise agglomerative hierarchical clustering (AHC) and spectral clustering (SPC). Based on the pipelines, we experiment all possible combinations of suggested techniques in Section 3: dimensionality reduction (DR), attention-based embedding aggregation (AA), and non-speech clustering (NS).

### 4.3. Results on each dataset

**DIHARD challenge dataset.** DIHARD [17, 18] is a series of challenges focusing on 'hard' diarisation, where the state-of-the-art systems tend to show poor performance. The data includes clinical interviews, child language acquisition recordings, restaurant recordings, and so on. We perform experiments on the evaluation sets of the first and the second challenge data.

We compare our pipeline to the winning systems of tracks 1, 2 from the challenge. Track 1 is speaker diarisation beginning from reference speech segment. Track 2 is diarisation from scratch. We analyze the effect of proposed techniques without disturbance by SAD error by comparing our results to the track 1 baseline. And we also show the performance of full pipeline including SAD by comparing track 2 results.

The results of each dataset are reported in Tables 1 and 2, respectively. All of the proposed techniques lead to performance improvement, and the system with all proposed embedding adaptation (+ DR + AA + NS) is our best configuration. In DIHARD I, our best configuration beats the winning systems in both tracks. In DIHARD

[1]https://github.com/nryant/dscore
[2]http://dihard.ldc.upenn.edu/competitions/73#results
[3]http://dihard.ldc.upenn.edu/competitions/74#results

Table 1: *Results on the DIHARD I challenge dataset using the baseline with every combination of proposed techniques. (FA: false alarm, MS: missed speech, SC: speaker confusion, SPC: Spectral clustering).*

| Configuration | DER | FA | MS | SC |
|---|---|---|---|---|
| **Oracle SAD (Track 1)** | | | | |
| SPC | 29.72 | 0.0 | 8.71 | 21.01 |
| SPC + DR | 29.16 | 0.0 | 8.71 | 20.45 |
| SPC + AA | 19.51 | 0.0 | 8.71 | 10.80 |
| SPC + NS | 27.55 | 0.0 | 8.71 | 18.83 |
| SPC + DR + AA | 19.12 | 0.0 | 8.71 | 10.41 |
| SPC + DR + NS | 20.30 | 0.0 | 8.71 | 11.59 |
| SPC + AA + NS | 17.24 | 0.0 | 8.71 | 8.52 |
| SPC + DR + AA + NS | **16.83** | 0.0 | 8.71 | 8.12 |
| AHC | 21.21 | 0.0 | 8.71 | 12.50 |
| AHC + DR | 19.85 | 0.0 | 8.71 | 11.14 |
| AHC + AA | 21.42 | 0.0 | 8.71 | 12.71 |
| AHC + NS | 20.97 | 0.0 | 8.71 | 12.26 |
| AHC + DR + AA | 19.50 | 0.0 | 8.71 | 10.78 |
| AHC + DR + NS | 18.45 | 0.0 | 8.71 | 9.74 |
| AHC + AA + NS | 21.59 | 0.0 | 8.71 | 12.88 |
| AHC + DR + AA + NS | 17.81 | 0.0 | 8.71 | 9.10 |
| Track 1 Winner [37] | 23.73 | - | - | - |
| **System SAD (Track 2)** | | | | |
| SPC | 49.20 | 16.17 | 10.60 | 22.41 |
| SPC + DR | 48.74 | 16.17 | 10.60 | 21.96 |
| SPC + AA | 37.01 | 16.17 | 10.60 | 10.22 |
| SPC + NS | 46.15 | 14.55 | 10.26 | 21.33 |
| SPC + DR + AA | 37.43 | 16.17 | 10.60 | 10.65 |
| SPC + DR + NS | 39.67 | 14.52 | 10.37 | 14.77 |
| SPC + AA + NS | **32.83** | 14.11 | 10.60 | 8.10 |
| SPC + DR + AA + NS | 33.54 | 13.66 | 11.61 | 8.27 |
| AHC | 40.24 | 16.17 | 10.60 | 13.45 |
| AHC + DR | 40.76 | 16.17 | 10.60 | 13.97 |
| AHC + AA | 39.46 | 16.17 | 10.60 | 12.68 |
| AHC + NS | 37.49 | 14.22 | 10.55 | 12.71 |
| AHC + DR + AA | 38.64 | 16.17 | 10.60 | 11.86 |
| AHC + DR + NS | 35.16 | 14.42 | 10.47 | 10.26 |
| AHC + AA + NS | 37.36 | 14.32 | 10.62 | 12.42 |
| AHC + DR + AA + NS | 33.44 | 13.82 | 11.34 | 8.28 |
| Track 2 Winner [38] | 35.51 | - | - | - |

II, it achieves a better score than the winner of track 1. Note that because our pipeline does not include an overlapping speech detector, only one speaker label is allocated for overlapping speech segments, evoking missed segments for overlapped segments in track 1.

**Internal dataset of real-world conversations.** This dataset comprises recordings collected across diverse domains such as informal discussions, offline meetings, and Zoom meetings. Each conversation is recorded under various conditions, from mobile phones to video conferencing microphone arrays, simulating real-world conditions. The recordings have been labeled professionally by trained annotators. Statistics of this dataset are reported in [30]. Table 3 gives the result of this dataset. The results are consistent with that of DIHARD I, II.

## 5. Conclusions

We propose three techniques to adapt embeddings for solving the diarisation problem based on the hypothesis that conventional approach of directly using speaker verification-oriented embeddings may not be ideal. DR reduces the dimensionality of embeddings, AA

Table 2: *Results on the DIHARD II challenge dataset using the baseline with every combination of proposed techniques. (FA: false alarm, MS: missed speech SC: speaker confusion, SPC: spectral clustering).*

| Configuration | DER | FA | MS | SC |
|---|---|---|---|---|
| **Oracle SAD (Track 1)** | | | | |
| SPC | 28.43 | 0.0 | 9.68 | 18.74 |
| SPC + DR | 28.61 | 0.0 | 9.68 | 18.93 |
| SPC + AA | 20.53 | 0.0 | 9.68 | 10.85 |
| SPC + NS | 26.77 | 0.0 | 9.68 | 17.08 |
| SPC + DR + AA | 19.70 | 0.0 | 9.68 | 10.02 |
| SPC + DR + NS | 21.08 | 0.0 | 9.68 | 11.39 |
| SPC + AA + NS | 18.84 | 0.0 | 9.68 | 9.15 |
| SPC + DR + AA + NS | **17.98** | 0.0 | 9.68 | 8.29 |
| AHC | 22.88 | 0.0 | 9.68 | 13.19 |
| AHC + DR | 20.68 | 0.0 | 9.68 | 10.99 |
| AHC + AA | 23.03 | 0.0 | 9.68 | 13.34 |
| AHC + NS | 22.56 | 0.0 | 9.68 | 12.87 |
| AHC + DR + AA | 20.10 | 0.0 | 9.68 | 10.41 |
| AHC + DR + NS | 18.79 | 0.0 | 9.68 | 9.10 |
| AHC + AA + NS | 23.14 | 0.0 | 9.68 | 13.45 |
| AHC + DR + AA + NS | 18.67 | 0.0 | 9.68 | 8.99 |
| Track 1 Winner[2] | 18.42 | - | - | - |
| **System SAD (Track 2)** | | | | |
| SPC | 51.33 | 19.17 | 11.67 | 20.49 |
| SPC + DR | 51.17 | 19.17 | 11.67 | 20.33 |
| SPC + AA | 41.10 | 19.17 | 11.67 | 10.26 |
| SPC + NS | 47.15 | 16.02 | 11.76 | 19.36 |
| SPC + DR + AA | 41.19 | 19.17 | 11.67 | 10.35 |
| SPC + DR + NS | 41.50 | 15.88 | 12.00 | 13.61 |
| SPC + AA + NS | 35.78 | 14.98 | 12.32 | 8.47 |
| SPC + DR + AA + NS | **35.67** | 14.28 | 13.46 | 7.91 |
| AHC | 44.15 | 19.17 | 11.67 | 13.31 |
| AHC + DR | 44.61 | 19.17 | 11.67 | 13.76 |
| AHC + AA | 43.98 | 19.17 | 11.67 | 13.14 |
| AHC + NS | 40.47 | 15.25 | 11.96 | 13.25 |
| AHC + DR + AA | 42.47 | 19.17 | 11.67 | 11.62 |
| AHC + DR + NS | 37.92 | 15.23 | 12.10 | 10.58 |
| AHC + AA + NS | 40.18 | 15.30 | 12.21 | 12.66 |
| AHC + DR + AA + NS | 35.78 | 14.55 | 13.18 | 8.04 |
| Track 2 Winner[3] | 27.11 | - | - | - |

Table 3: *Results on the internal conversations dataset using the baseline with every combination of proposed techniques. (FA: false alarm, MS: missed speech, SC: speaker confusion).*

| Configuration | DER | FA | MS | SC |
|---|---|---|---|---|
| AHC | 24.67 | 4.69 | 4.47 | 15.51 |
| AHC + DR + AA + NS | **17.97** | 2.18 | 4.28 | 11.52 |
| Baseline from [30] | 45.9 | 3.1 | 26.4 | 16.4 |
| Best system from [30] | 30.0 | 2.2 | 6.9 | 20.9 |

aggregates embeddings refining the affinity matrix, and NS enables consideration of non-speech regions. All three techniques can be independently adopted to the existing process pipeline where applying all three techniques improved the performance the most. We find that these proposed techniques demonstrate consistent improvement regardless of both SAD and clustering frameworks.

# 6. References


[1] T. J. Park, N. Kanda, D. Dimitriadis, K. J. Han, S. Watanabe, and S. Narayanan, "A review of speaker diarization: Recent advances with deep learning," *arXiv preprint arXiv:2101.09624*, 2021.

[2] S. Horiguchi, N. Yalta, P. Garcia, Y. Takashima, Y. Xue, D. Raj, Z. Huang, Y. Fujita, S. Watanabe, and S. Khudanpur, "The hitachi-jhu dihard iii system: Competitive end-to-end neural diarization and x-vector clustering systems combined by dover-lap," *arXiv preprint arXiv:2102.01363*, 2021.

[3] Y. Xue, S. Horiguchi, Y. Fujita, Y. Takashima, S. Watanabe, P. Garcia, and K. Nagamatsu, "Online end-to-end neural diarization handling overlapping speech and flexible numbers of speakers," *arXiv preprint arXiv:2101.08473*, 2021.

[4] S. Horiguchi, P. Garcia, Y. Fujita, S. Watanabe, and K. Nagamatsu, "End-to-end speaker diarization as post-processing," *arXiv preprint arXiv:2012.10055*, 2020.

[5] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *Proc. ICASSP*. IEEE, 2017, pp. 4930–4934.

[6] K. J. Han, S. Kim, and S. S. Narayanan, "Strategies to improve the robustness of agglomerative hierarchical clustering under data source variation for speaker diarization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1590–1601, 2008.

[7] N. Ryant, P. Singh, V. Krishnamohan, R. Varma, K. Church, C. Cieri, J. Du, S. Ganapathy, and M. Liberman, "The third dihard diarization challenge," *arXiv preprint arXiv:2012.01477*, 2020.

[8] S. Watanabe, M. Mandel, J. Barker, and E. Vincent, "Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," *arXiv preprint arXiv:2004.09249*, 2020.

[9] M. Diez, L. Burget, F. Landini, S. Wang, and H. Černockỳ, "Optimizing bayesian hmm based x-vector clustering for the second dihard speech diarization challenge," in *Proc. ICASSP*. IEEE, 2020, pp. 6519–6523.

[10] M. Diez, L. Burget, S. Wang, J. Rohdin, and J. Cernockỳ, "Bayesian hmm based x-vector clustering for speaker diarization." in *Proc. Interspeech*, 2019, pp. 346–350.

[11] M. Diez, L. Burget, and P. Matejka, "Speaker diarization based on bayesian hmm with eigenvoice priors." in *Proc. Odyssey*, 2018, pp. 147–154.

[12] I. Medennikov, M. Korenevsky, T. Prisyach, Y. Khokhlov, M. Korenevskaya, I. Sorokin, T. Timofeeva, A. Mitrofanov, A. Andrusenko, I. Podluzhny *et al.*, "Target-speaker voice activity detection: a novel approach for multi-speaker diarization in a dinner party scenario," in *Proc. Interspeech*, 2020.

[13] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Science and Language*, 2019.

[14] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. Interspeech*, 2018.

[15] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *Proc. Interspeech*, 2017.

[16] F. Landini, O. Glembek, P. Matějka, J. Rohdin, L. Burget, M. Diez, and A. Silnova, "Analysis of the but diarization system for voxconverse challenge," *arXiv preprint arXiv:2010.11718*, 2020.

[17] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "First dihard challenge evaluation plan," *2018, tech. Rep.*, 2018.

[18] ——, "The second dihard diarization challenge: Dataset, task, and baselines," in *Proc. Interspeech*, 2019.

[19] J. S. Chung, J. Huh, A. Nagrani, T. Afouras, and A. Zisserman, "Spot the conversation: speaker diarisation in the wild," in *Proc. Interspeech*, 2020.

[20] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The ami meeting corpus: A pre-announcement," in *International workshop on machine learning for multimodal interaction*. Springer, 2005, pp. 28–39.

[21] A. B. Johnston and D. C. Burnett, *WebRTC: APIs and RTCWEB protocols of the HTML5 real-time web*. Digital Codex LLC, 2012.

[22] Y. Kwon, H.-S. Heo, B.-J. Lee, and J. S. Chung, "The ins and outs of speaker recognition: lessons from voxsrc 2020," in *Proc. ICASSP*, 2021.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[24] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," in *Proc. Interspeech*, 2020.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. ECCV*. Springer, 2016, pp. 630–645.

[26] H. S. Heo, B.-J. Lee, J. Huh, and J. S. Chung, "Clova baseline system for the VoxCeleb speaker recognition challenge 2020," *arXiv preprint arXiv:2009.14153*, 2020.

[27] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. CVPR*, 2019, pp. 4690–4699.

[28] W. H. Day and H. Edelsbrunner, "Efficient algorithms for agglomerative hierarchical clustering methods," *Journal of classification*, vol. 1, no. 1, pp. 7–24, 1984.

[29] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[30] Y. Kwon, H. S. Heo, J. Huh, B.-J. Lee, and J. S. Chung, "Look who's not talking," in *Proc. SLT*, 2021.

[31] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[32] 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.

[33] H.-S. Heo, J.-w. Jung, Y. Kwon, J. Kim, J. Huh, J. S. Chung, and B.-J. Lee, "NAVER CLOVA SUBMISSION TO THE THIRD DIHARD CHALLENGE," Tech. Rep., 2021.

[34] X. Wu, R. He, Z. Sun, and T. Tan, "A light cnn for deep face representation with noisy labels," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2884–2896, 2018.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017.

[36] J. Cheng, L. Dong, and M. Lapata, "Long short-term memory-networks for machine reading," 2016, pp. 551–561.

[37] 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." in *Proc. Interspeech*, 2018, pp. 2808–2812.

[38] F. Landini, S. Wang, M. Diez, L. Burget, P. Matějka, K. Žmolíková, L. Mošner, O. Plchot, O. Novotnỳ, H. Zeinali *et al.*, "But system description for dihard speech diarization challenge 2019," *arXiv preprint arXiv:1910.08847*, 2019.