



# Online Speaker Diarization with Core Samples Selection

Yanyan Yue<sup>1</sup>, Jun Du<sup>1,\*</sup>, Mao-Kui He<sup>1</sup>, Yu Ting Yeung<sup>2</sup>, Renyu Wang<sup>2</sup>

<sup>1</sup>University of Science and Technology of China, HeFei, China

<sup>2</sup>Huawei Noah's Ark Lab

yyyue@mail.ustc.edu.cn, jundu@ustc.edu.cn, hmk1754@mail.ustc.edu.cn,  
yeung.yu.ting@huawei.com, wangrenyu1@huawei.com

## Abstract

We propose a novel online speaker diarization approach based on the VBx algorithm which works well on the offline speaker diarization tasks. To efficiently process long-time recordings, we perform the online diarization in a block-wise manner. First, we devise a core samples updating strategy utilizing time penalty function, which can preserve important historical information with a low memory cost. Then we select clustering samples from core samples by stratified sampling to enhance the variability among samples and retain sufficient speaker identity information, which helps VBx to improve classification accuracy on a small amount of data. Finally, we solve the label ambiguity problem by a global constrained clustering algorithm. We evaluate our system on DIHARD and AMI datasets. The experimental results demonstrate that our online approach achieves superior performance compared with the state-of-the-art.

**Index Terms:** online speaker diarization, VBx, core samples, constrained clustering

## 1. Introduction

Speaker diarization is a task of classifying recordings into homogeneous speaker-specific regions, i.e. "who spoke when" [1]. Good speaker diarization results play an important role in applications such as speech transcription, dominant speaker detection, speech indexing and meeting summary [2, 3].

The existing literature on speaker diarization is extensive and mainly focuses on speaker embedding clustering algorithms [4, 5]. These clustering-based techniques mainly consist of modules such as speech segmentation, speaker embedding extraction [6, 7, 8, 9], and clustering [10, 11, 12, 13]. The speaker embedding extraction module often employs i-vector [6, 7] and some neural network-based embeddings, such as x-vector [8] and d-vector [9]. The commonly used clustering algorithms including agglomerative hierarchical clustering (AHC) [10], k-means [11], and spectral clustering (SC) [12]. Among them, variational Bayesian hidden Markov model with x-vector (VBx) [13] achieved compelling performance and ranked first in DIHARD-II Challenge [14]. However, the problem of overlapping speech is the pain point of clustering-based diarization methods and has not been effectively dealt with due to the hard clustering. Recently, end-to-end neural speaker diarization (EEND) [15, 16, 17] and target-speaker speech activity detection (TS-VAD) [18] have been proposed to better handle the speaker overlap regions.

The methods discussed above are all offline methods. There are still relatively little studies on online speaker diarization. Compared with the offline diarization, the online speaker diarization is more challenging which needs to assign speakers

in the arriving recordings on the fly. A superior online system should correctly classify speakers and be able to detect emerging speakers in real time. There is always a tradeoff between the high accuracy and low latency in online diarization task.

Some of the early research have been conducted based on GMM-UBM systems [19, 20, 21]. Speaker adaptation is a joint denominator of these online systems. In [22], incremental learning was applied to a VBHMM-based speaker classification system to iteratively update the model parameters with the online EM algorithm. The graph-based reclustering process [23] was also designed to improve the performance of chkpt-AHC online diarization system. In addition to online system based on traditional clustering methods, Google proposed a fully supervised framework UIS-RNN [24], which can obtain better results than unsupervised online system when annotation data are available. Subsequently, the UIS-RNN was modified by proposing a new loss function and speaker turn modeling [25] to improve the performance. Furthermore, a common idea for online diarization is to perform clustering from scratch whenever new data arrive. In [26], the X-means was utilized to perform clustering, which is more efficient than k-means. An update mechanism was also designed to modify historical errors. However, this method has its limitation in long-time recordings, so some studies use a block-wise process approach, where clustering is conducted on a limited number of segments at a time. In [27], Coria et al. proposed end-to-end speaker segmentation neural network on a sliding window of five-second long to perform segmentation, then obtained the final labels by incremental clustering. In [28, 29], buffer mechanism was used to apply EEND to online systems. The authors compared four strategies for selecting data to fill the buffer, and yielded comparable results with the baseline system on DIHARD-II. Both systems are capable of dealing with overlapping speech.

In this paper, we try to solve the challenging online speaker diarization task. First, we use VBx, which is robust and works well in offline tasks, as a baseline to build the online system by block-wise processing. As most of the current online diarization algorithms are based on the improvements of existing offline algorithms, and predictably, online systems always track offline algorithms in terms of performance. Then, unlike [27], [29], we propose a new selection strategy to keep the most representative  $N$  segments of each class in the historical contexts, named core samples. Whenever new data arrive,  $L$  samples are drawn from the core samples in a stratified sampling manner and sent to VBx with the new sample to perform clustering. Finally, due to the unsupervised manner in VBx, the label ambiguity problem occurs. Therefore, we utilize the constrained clustering to solve the label matching problem. We conducted experiments on the widely used real datasets, namely DIHARD and AMI, and achieved comparable or even better diarization results than offline systems.

\*corresponding author

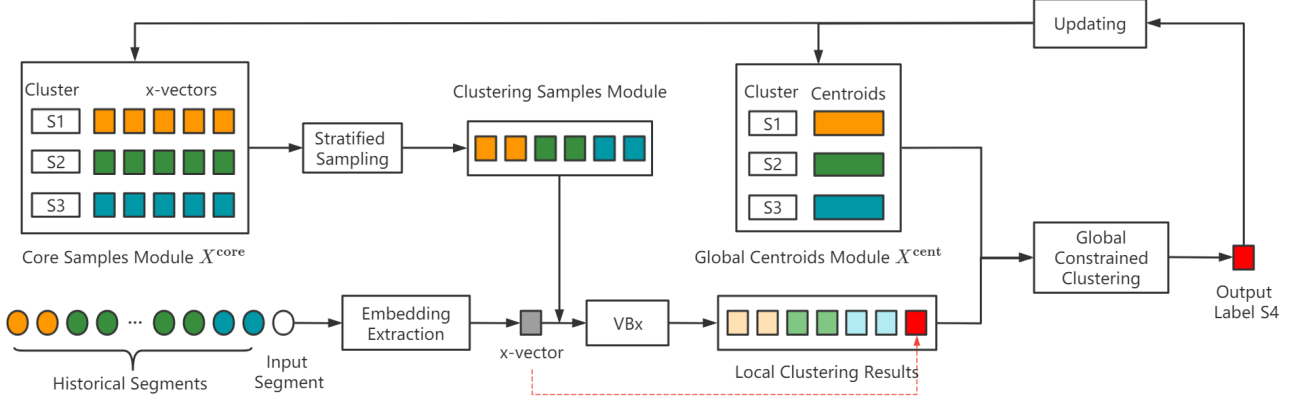


Figure 1: The pipeline of our online diarization system.

## 2. Proposed Method

### 2.1. Overview

Most of the online diarization systems are based on improvement of existing offline algorithms, such as GMM-UBM [20, 21], EEND [29], and depend on offline algorithms in terms of performance. Specially, VBx is quite robust and performs well on offline speaker diarization tasks. However, there is no literature on developing online systems based on VBx so far. Different from other VBHMM-based systems, VBx only applies a standard normal prior distribution to the hidden variables. VBx does not assume a prior distribution for the transition probabilities, limiting the modification of VBx to online manner, which is similar to [22]. Towards this end, we use a block-wise technique to construct an online system to take advantage of high classification accuracy of VBx.

The whole framework is illustrated in Fig 1. Our system contains three dynamic memory modules: the core samples module, the clustering samples module, and the global centroids module. The core samples module stores some of the most representative samples of each category. The clustering samples module stores the  $L$  samples drawn from the core samples module. The global centroids module stores the center of each category in historical data. Although VBx contains a time-consuming resegmentation module, VBx can still meet real-time requirements because of executing on a small amount of data at a time. Furthermore, benefiting from the core samples module, we do not have to store all the historical data, which reduces the memory cost. Similar to other unsupervised clustering-based algorithms, the online VBx system suffers from label ambiguity problem. We address this issue by global constrained clustering method. In addition, we design a core sample update mechanism based on time penalty function, which further improves the performance. Details of the proposed system are described in the following subsections.

### 2.2. Core Samples Updating

In the block-wise process with a fixed block size, the classification accuracy of VBx on fewer data is decisive for the final classification result. Obviously, VBx has a high classification accuracy when the data distribution of different categories in the block is relatively different. A natural idea is to use a sliding window to select samples in blocks, but it is difficult to include sufficient speaker identity information. In contrast, di-

rectly selecting samples from historical samples to capture sufficient permutation information requires maintaining all historical data, which increases memory cost as speech duration increases. And the quality of the selected samples may be uneven. Therefore, we aim to ensure that the data within a block contain sufficient speaker identity information while data in the same class are as aggregated as possible. The different classes are as far away from each other as possible. To this end, we propose a dynamic memory module to retain the most representative samples of each category (namely, the speaker identity), called the core samples. In the following, we will describe the process of constructing and updating the core samples in real time.

We set the maximum allowed number of each class in the core samples to  $N$ . After processing some of the data, we have global centroids,  $\mathbf{X}^{cent} = \{\mathbf{X}_1^{cent}, \mathbf{X}_2^{cent}, \dots, \mathbf{X}_C^{cent}\}$ , where  $C$  is the number of speakers that have appeared.  $\mathbf{X}_i^{cent}$  is obtained by averaging all embeddings contained in cluster  $i$ . And core samples  $\mathbf{X}^{core} = \{\mathbf{X}_1^{core}, \mathbf{X}_2^{core}, \dots, \mathbf{X}_C^{core}\}$ ,  $\mathbf{X}_c^{core} = \{\mathbf{x}_{c,i_1}^{core}, \mathbf{x}_{c,i_2}^{core}, \dots, \mathbf{x}_{c,i_N}^{core}\}$ , where  $i_k$  is the core sample index in the historical data. If the new sample  $\mathbf{x}_t$  belongs to class  $c$ , add  $\mathbf{x}_t$  to  $\mathbf{X}_c^{core}$ . Supposing that the sample number of class  $c$  in the core samples is greater than  $N$  at this point, the core sample  $\mathbf{x}_{c,i}^{core}$  with the lowest score in  $\mathbf{X}_c^{core}$  is to be deleted. The common score is calculated by the cosine similarity between the core sample  $\mathbf{x}_{c,i}^{core}$  and the center of the corresponding class  $\mathbf{X}_c^{cent}$ :

$$d(c, i) = \cos(\mathbf{x}_i, \mathbf{X}_c^{cent}) \quad (1)$$

This score function does not take the time decay into account. In fact, the accuracy of the classification results of the samples that appear earlier in  $\mathbf{X}_c^{core}$  is relatively low, and the impact on the future samples is also relatively small. Accordingly, we design a time penalty function  $f_t(i)$  according to the proximity between  $\mathbf{x}_{c,i}^{core}$  and  $\mathbf{x}_t$ . We believe that all samples in  $\mathbf{X}_c^{core}$  whose distance from the new sample  $\mathbf{x}_t$  is greater than  $N_{invl}$  have the same impact on the new sample. Here, ‘distance’ refers to the difference between the time sampling indices. Therefore, the time penalty function  $f_t(i)$  is designed as a piece-wise function:

$$f_t(i) = \begin{cases} 1 & t - i \leq N_{invl} \\ \lambda & t - i > N_{invl} \end{cases} \quad (2)$$

Finally, our score function considers the effects of both spatial and temporal proximity. At time  $t$ , the score of the core sample  $\mathbf{x}_{c,i}^{core}$  is expressed as:

$$d_t(c, i) = d(c, i) * f_t(i) \quad (3)$$

### 2.3. Stratified Sampling

As mentioned earlier, in addition to ensuring the discrepancies of the speaker identities in blocks, the clustering samples should also contain sufficient identity information. Therefore, we select  $L$  samples from the core samples by stratified sampling to fill in the clustering samples. In the case of a large number of categories and sizable differences between categories, using stratified sampling can better draw representative samples. Compared with the standard stratified sampling, there are two modifications in the proposed method. On one hand, when the number of samples of a certain class in the core samples is very small, we do not choose according to its proportion, but all of them. This can avoid serious category imbalance in the clustering samples. Given core samples  $\mathbf{X}^{\text{core}} = \{\mathbf{X}_1^{\text{core}}, \mathbf{X}_2^{\text{core}}, \dots, \mathbf{X}_C^{\text{core}}\}$ , here we assume that the number of each class is  $n_c$ , and  $n_1 \leq n_2 \leq \dots \leq n_C$ . If  $\sum_{i=1}^C n_i \leq L$ , then all data in core samples are fed into clustering sample. Otherwise, the number of samples that should be selected for each category is

$$l_i = \begin{cases} n_i & i < C \text{ and } n_i \leq n_{\min} \\ p_i & i < C \text{ and } n_i \geq n_{\min} \\ L - \sum_{i=1}^{C-1} l_i & i = C \end{cases} \quad (4)$$

where  $p_i = \lceil \frac{n_i}{\sum_{i=1}^C n_i} \times L \rceil$ . Here  $\lceil \cdot \rceil$  means rounding up.  $n_{\min}$  is the minimum number of each category allowed to be selected by all. Considering the importance of the samples closer to the new sample, we pick the latest  $l_i$  samples instead of picking them randomly. It is noteworthy that the selected samples are arranged according to their relative order in the historical data. The selected samples are then concatenated with the new sample and finally fed into the VBx classifier for classification.

### 2.4. Global Constrained Clustering

After we obtain the initial label of the new sample by VBx, we cannot determine whether the new sample belongs to an existing class or from a new speaker due to the label ambiguity problem. Inspired by [30], we solve this problem by a simple global constrained clustering algorithm. Given the classification result of VBx classifier, we calculate the local centroids,  $\mathbf{X}^{\text{VBx}} = \{\mathbf{X}_1^{\text{VBx}}, \mathbf{X}_2^{\text{VBx}}, \dots, \mathbf{X}_{C'}^{\text{VBx}}\}$ .  $\mathbf{X}_{c'}^{\text{VBx}}$  is the average of the x-vector of cluster  $c'$  in the block. Using local and global centroids to construct similarity matrix  $\mathbf{D} \in \mathbb{R}^{C \times C'}$ , whose elements are defined as:

$$d_{ij} = \cos(\mathbf{X}_i^{\text{cent}}, \mathbf{X}_j^{\text{VBx}}) \quad (5)$$

We match the local category with the global category by the similarity matrix. It should be noted that different classes in the local category cannot be matched to the same class in the global category. To achieve this goal, we define a weight matrix  $\mathbf{W}$  with elements initialized as ones. The matching process mainly consists of two steps: maximum value matching and weight updating. First, we obtain the most likely pair of indices

$$(l, k) = \arg \max_{i,j} (\mathbf{D} \odot \mathbf{W}) \quad (6)$$

where the samples in local cluster  $k$  most likely belongs to global cluster  $l$ . Here  $\odot$  denotes element wise product between two matrices. Then we update weight matrix  $\mathbf{W}$

$$\mathbf{W}[l, :] = \mathbf{W}[:, k] = 0 \quad (7)$$

Through Eq.(7), different speakers in the VBx output are not matched to the same global label. Repeat the above two steps until the similarity matrix becomes a zero matrix. If the class of a new sample does not match the global class at the end of the matching process, the new sample belongs to a new speaker.

## 3. Experiments

### 3.1. Experimental Setup

We evaluate diarization error rates (DER) [31] on DIHARD-II [14], DIHARD-III [32] and AMI datasets [33]. The DIHARD-II and DIHARD-III datasets contain 11 subsets with audio duration of 5-10 minutes. The AMI corpus is a widely used conference corpus with longer duration, in average half an hour per audio. We divide the AMI dataset into train, dev, and eval sets using the division criteria in [34].

Clustering is built on speech segments. We set the length of overlapping sliding window to be 2s and the overlap ratio is 50%. Whenever a two-second speech segment arrives, we extract an embedding for it. We use the same recipe published by BUT speech team<sup>1</sup> to extract x-vectors. In the offline VBx model, the parameter controlling iteration stop  $\epsilon$  is set to 1e-5. In the online model, we found that setting  $\epsilon$  to 0 is able to get results comparable to 1e-5 while taking less time. The maximum number  $N$  of storage allowed for each class in core samples is set to 120. We use the development datasets to tune other parameters and evaluate our systems on the evaluation datasets. For all the datasets, oracle boundary information is applied.

### 3.2. Ablation Experiments

Table 1 compares the impact of four selection strategies and two label matching methods on the performance of the online system at different block sizes. We carry out experiments with three block sizes of 60, 90, 120 segments. ‘Sliding Window’ means using the latest  $L$  segments to classify with new segments each time. ‘Historical Samples’ indicates stratified sampling from all historical data as mentioned in Section 2.3. The third and fourth lines are all stratified sampling from the core samples, and the difference lies in the criterion for updating core samples. The third line does not consider time decay and updates the core samples based on cosine similarity only. The fourth line uses piece-wise time penalty. All four systems use global constrained clustering for label matching. The fifth line indicates that instead of using global constrained clustering, label matching is performed using the reconciliation algorithm used in [26]. From this table we can make several observations. First, the performance of online diarization is positively correlated with the block size regardless of the selection strategy. This is consistent with our empirical knowledge, i.e., when keep the chosen clustering algorithm, the more information we can utilize in classification, the more accurate the classification results will be. Second, our proposed method can efficiently process the long-time recordings. By comparing the different selection strategies, using core samples with piece-wise time penalty has achieved the best performance on both DIHARD-III and AMI evaluation sets. It is worth noting that the best results of online system on AMI are very close to offline VBx results. However, on the DIHARD-III evaluation set, the performance improvements brought by core samples are not as significant as those on AMI. There is a disparity with offline VBx. This is because DIHARD-III contains recordings from multiple

<sup>1</sup><https://github.com/BUTSpeechFIT/VBx>

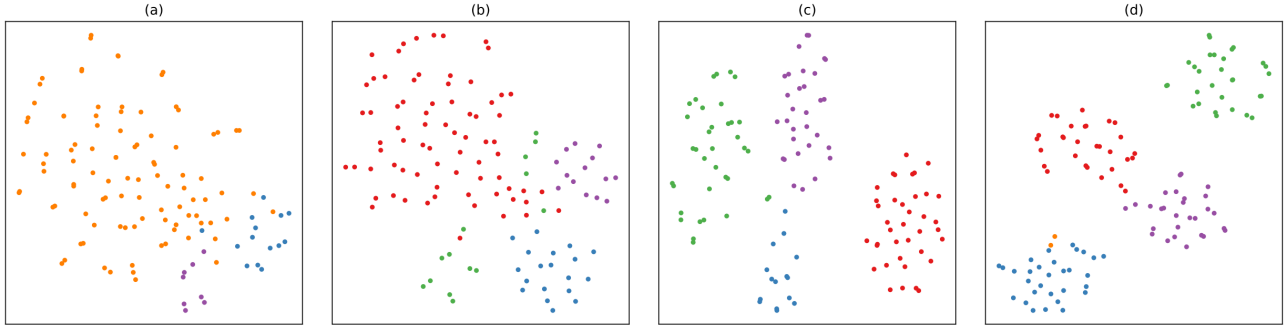


Figure 2: Visualization of  $L$  samples selected by four different selection strategies: (a) sliding window, (b) stratified sampling from historical data, (c) core samples without time penalty, and (d) core samples with piece-wise time penalty.

Table 1: DERs(%) of offline VBx and online systems with different selection strategies and block sizes on AMI and DIHARD-III evaluation sets.

System	Online					
	AMI			DIHARD-III		
	N=60	N=90	N=120	N=60	N=90	N=120
Sliding Window	23.8	21.9	21.3	21.8	20.6	20.0
Historical Samples	23.3	20.6	20.1	21.0	20.1	19.7
Core Samples+Eq.(1)	23.7	22.7	22.1	20.7	19.9	19.6
Core Samples+Eq.(3)	<b>22.9</b>	<b>19.7</b>	<b>19.0</b>	<b>20.6</b>	<b>19.9</b>	<b>19.3</b>
-w/o Constrained	23.3	20.0	19.5	21.7	20.3	19.9
Offline VBx	18.3			15.7		

complex domains, resulting in poor performance of the online system. Furthermore, selecting core samples requires sufficient historical data, but the audio duration in DIHARD-III dataset is relatively short. Third, the global constrained clustering can deal with label matching problem well, yielding better results than reconciliation algorithm. The reason is the reconciliation algorithm in [26] only performs label matching on the selected blocks and does not make use of the global information.

In order to intuitively demonstrate the differences among sample selection strategies, we select a recording in the AMI evaluation set and visualize the 120 segments selected by the four selection strategies at the same time index in Figure 2. Each dot in the figure represents a x-vector based segment. Different colors represent different speakers in each subfigure. Figure 2 demonstrates that the segments selected from the sliding window do not contain sufficient speaker identity information. The samples selected from the core samples with time penalty contain more differences than those directly obtained from historical data. What’s more, the number of different categories is more balanced, which may help to achieve better performance in our system.

### 3.3. Results for Different Diarization System

To verify the validity of the proposed method, we compare our method with other existing results on DIHARD-II evaluation set in Table 2. Here, we select clustering samples from the core samples and update core samples with Eq.(3), where  $\lambda$  is set to 0.2 and  $N_{invl}$  is 30. The proposed online system achieves a DER of 23.1%, which outperforms the UIS-RNN-SML and FLEX-STB. Moreover, compared with DIHARD-II official baseline, our approach yields a 11.2% relative diarization error rate reduction. However, similar to the results on DIHARD-III, there

Table 2: DERs(%) results on DIHARD-II evaluation set.

System	Method	DER
Offline	DIHARD-II Baseline [14]	26.0
	VBx [34]	18.6
Online	UIS-RNN-SML [25]	27.3
	FLEX-STB [29]	25.8
	Proposed	23.1

is still a large performance gap compared with offline VBx. The results indicate that the application of online speaker diarization in difficult scenarios is still in its infancy.

### 3.4. Real Time Analysis

Our experiments were performed on one NVIDIA Geforce RTX 3090 GPU. To analyze the real-time performance of our online system, we calculate the average time required to process a segment. This is obtained by dividing the time to process the entire dataset by the total number of segments. Here each segment is 2s. The average time per segment is 0.2s on AMI dataset and 0.25s on DIHARD dataset. This delay is acceptable for an online system.

## 4. Conclusions

In this paper, we propose an online speaker diarization method that can handle long-time recordings. Specifically, we exploit the high classification accuracy of VBx through a block-wise approach and introduce a time-decay-based core sample selection mechanism to further improve the classification performance of VBx on a small amount of data. In addition, we use global constrained clustering to solve the label matching problem. Moreover, our approach can also be expanded to other offline clustering algorithms. Experimental results on DIHARD and AMI datasets suggest that our online system can efficiently process the long-time recordings. Future work will focus on handling overlap regions in an end-to-end manner.

## 5. Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No. 62171427. We also acknowledge Weixiang Hu, Yu Lu, Baohui Wang and Yingying Wang of Huawei Consumer Business Group for useful suggestions.

## 6. References

- [1] T. J. Park, N. Kanda, D. Dimitriadis, K. Han, S. Watanabe, and S. Narayanan, "A review of speaker diarization: Recent advances with deep learning," *Computer Speech & Language*, vol. 72, p. 101317, 2022.
- [2] D. Vijayasenan, F. Valente, and H. Bourlard, "An information theoretic approach to speaker diarization of meeting data," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 7, pp. 1382–1393, 2009.
- [3] 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.
- [4] 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, no. 10, pp. 2015–2028, 2013.
- [5] G. Sell and D. Garcia-Romero, "Speaker diarization with PLDA i-vector scoring and unsupervised calibration," in *2014 IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 413–417.
- [6] 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.
- [7] S. Madikeri, I. Himawan, P. Motlicek, and M. Ferras, "Integrating online i-vector extractor with information bottleneck based speaker diarization system," *Idiap*, 2015.
- [8] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.
- [9] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4879–4883.
- [10] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 4930–4934.
- [11] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with LSTM," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5239–5243.
- [12] Q. Lin, R. Yin, M. Li, H. Bredin, and C. Barras, "LSTM based similarity measurement with spectral clustering for speaker diarization," in *Interspeech 2019*, 2019, pp. 366–370.
- [13] M. Diez, L. Burget, F. Landini, and J. Černocký, "Analysis of speaker diarization based on bayesian HMM with eigenvoice priors," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 355–368, 2020.
- [14] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "The second DIHARD diarization challenge: Dataset, task, and baselines," 2019. [Online]. Available: <https://arxiv.org/abs/1906.07839>
- [15] Y. Fujita, N. Kanda, S. Horiguchi, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with permutation-free objectives," 2019. [Online]. Available: <https://arxiv.org/abs/1909.05952>
- [16] Y. Fujita, N. Kanda, S. Horiguchi, Y. Xue, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with self-attention," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 296–303.
- [17] S. Horiguchi, Y. Fujita, S. Watanabe, Y. Xue, and K. Nagamatsu, "End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attractors," 2020. [Online]. Available: <https://arxiv.org/abs/2005.09921>
- [18] I. Medennikov, M. Korenevsky, T. Prisyach, Y. Khokhlov, M. Korenevskaya, I. Sorokin, T. Timofeeva, A. Mitrofanov, A. Andrusenko, I. Podluzhny, A. Laptev, and A. Romanenko, "Target-speaker voice activity detection: A novel approach for multi-speaker diarization in a dinner party scenario," in *Interspeech 2020*, 2020, pp. 274–278.
- [19] K. Markov and S. Nakamura, "Never-ending learning system for on-line speaker diarization," in *2007 IEEE Workshop on Automatic Speech Recognition Understanding (ASRU)*, 2007, pp. 699–704.
- [20] J. T. Geiger, F. Wallhoff, and G. Rigoll, "GMM-UBM based open-set online speaker diarization," in *Interspeech 2010*, 2010, pp. 2330–2333.
- [21] C. Vaquero, O. Vinyals, and G. Friedland, "A hybrid approach to online speaker diarization," in *Interspeech 2010*, 2010, pp. 2638–2641.
- [22] T. Koshinaka, K. Nagatomo, and K. Shinoda, "Online speaker clustering using incremental learning of an ergodic hidden Markov model," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 4093–4096.
- [23] Y. Zhang, Q. Lin, W. Wang, L. Yang, X. Wang, J. Wang, and M. Li, "Low-latency online speaker diarization with graph-based label generation," 2021. [Online]. Available: <https://arxiv.org/abs/2111.13803>
- [24] A. Zhang, Q. Wang, Z. Zhu, J. Paisley, and C. Wang, "Fully supervised speaker diarization," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6301–6305.
- [25] E. Fini and A. Brutti, "Supervised online diarization with sample mean loss for multi-domain data," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7134–7138.
- [26] D. Dimitriadis and P. Fousek, "Developing on-line speaker diarization system," in *Interspeech 2017*, 2017, pp. 2739–2743.
- [27] J. M. Coria, H. Bredin, S. Ghannay, and S. Rosset, "Overlap-aware low-latency online speaker diarization based on end-to-end local segmentation," 2021. [Online]. Available: <https://arxiv.org/abs/2109.06483>
- [28] Y. Xue, S. Horiguchi, Y. Fujita, S. Watanabe, P. García, and K. Nagamatsu, "Online end-to-end neural diarization with speaker-tracing buffer," in *2021 IEEE Spoken Language Technology Workshop (SLT)*, 2021, pp. 841–848.
- [29] Y. Xue, S. Horiguchi, Y. Fujita, Y. Takashima, S. Watanabe, P. Garcia, and K. Nagamatsu, "Online streaming end-to-end neural diarization handling overlapping speech and flexible numbers of speakers," 2021. [Online]. Available: <https://arxiv.org/abs/2101.08473>
- [30] P. Kulshreshtha and T. Guha, "An online algorithm for constrained face clustering in videos," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 2670–2674.
- [31] R. Vs, "The 2009 (RT-09) rich transcription meeting recognition evaluation plan," 2009.
- [32] N. Ryant, P. Singh, V. Krishnamohan, R. Varma, K. Church, C. Cieri, J. Du, S. Ganapathy, and M. Liberman, "The third DIHARD diarization challenge," 2020. [Online]. Available: <https://arxiv.org/abs/2012.01477>
- [33] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner, "The AMI meeting corpus: A pre-announcement," in *Machine Learning for Multimodal Interaction*, 2006, pp. 28–39.
- [34] F. Landini, J. Profant, M. Diez, and L. Burget, "Bayesian HMM clustering of x-vector sequences (VBx) in speaker diarization: Theory, implementation and analysis on standard tasks," *Computer Speech & Language*, vol. 71, p. 101254, 2021.