Selective Channel Attention based Target Speaker Voice Activity Detection for Speaker Diarization under AD-HOC Microphone Array Settings

Hongyu Zhang[1], *Ming Cheng*[1], *Jing Feng*[2], *Ming Li*[1,3]

[1]School of Computer Science, Wuhan University, China
[2]School of Artificial Intelligence, Wuhan University, China
[3]Suzhou Municipal Key Laboratory of Multimodal Intelligent Systems, Digital Innovation Research Center, Duke Kunshan University, China

gfeng@whu.edu.cn, ming.li369@dukekunshan.edu.cn

Abstract

Speaker diarization benefits from multi-channel microphone arrays, yet current systems struggle with diverse configurations. We address this by simulating a dataset with various microphone topologies and proposing Selective Channel Attention-based Target Speaker Voice Activity Detection (SCA-TSVAD). We utilize cross-channel self-attention with masking mechanisms to enable selective attention on specific channels, allowing for the effective processing of audio data with variable multi-channel configurations. SCA-TSVAD is built upon the foundation of single-channel TSVAD. It performs superior on our simulated dataset, showcasing its robustness across diverse array configurations. To further validate the effectiveness of a real dataset, we evaluate SCA-TSVAD on the real-world Ali-Meeting database, where it successfully handles multi-channel audio inputs even when some channels were unavailable or malfunctioning, proving its practical applicability.

Index Terms: Target-speaker voice activity detection, Multi-channel speaker diarization, Selective channel attention

1. Introduction

Speaker diarization, often called the "who spoke when" task, involves identifying and segmenting speech in multi-party conversations [1] by attributing each segment to the corresponding speaker. It consists of determining the temporal boundaries of each speaker's utterances and labeling them with speaker identifiers. Traditional clustering-based speaker diarization systems [2, 3], which assume single-speaker dominance in each segment, struggle to handle overlapped speech without additional modules. Previous research has attempted to address this limitation through various approaches, including speech separation [4] as a pre-processing step, target-speaker voice activity detection (TS-VAD) [5] as post-processing, and end-to-end neural diarization (EEND) [6, 7] for overlap-aware diarization.

In contrast, multi-channel speech processing systems [8], can leverage spatial information to enhance source discrimination. Among the diverse designs of such systems, cross-channel attention has emerged as a powerful tool, demonstrating success in tasks such as speech enhancement [9], separation [10, 11], recognition [12], and diarization [13].

Most existing systems, while promising, are tailored for specific array setups, limiting their adaptability to diverse microphone arrangements. This lack of flexibility hinders practical deployment, as real-world applications often involve varying microphone topologies. Furthermore, practical issues such as structural damage or loose connections can lead to partial channel loss even in fixed microphone array setups. In speech separation, [11] tackles issues such as unknown microphone counts and geometries, as well as asynchronous signals, by employing a Transformer-based architecture to model the signals from spatially distributed microphones. Similarly, in speaker verification, [14] proposes an attention-based multi-channel system for ad-hoc microphone arrays. The method employs an inter-channel processing layer with residual self-attention to weigh different microphones and a global fusion layer to integrate signals independently of channel numbers.

This paper proposes the Selective Channel Attention-based Target Speaker Voice Activity Detection (SCA-TSVAD), which extends the single-channel TS-VAD framework [7] by incorporating selective channel attention for variable channels. We first construct a simulated multi-channel, multi-speaker dataset with various microphone configurations based on LibriSpeech [15]. Our code about the data simulation, as well as the database and examples, is publicly available[1]. The SCA-TSVAD model demonstrates excellent performance on this simulated dataset, showcasing its robustness and adaptability to different microphone configurations. We further validate our approach on the real-world Ali-Meeting dataset, confirming its effectiveness in handling incomplete or faulty multi-channel inputs.

2. Methods

2.1. Selective Channel Attention-based TSVAD

This section describes the framework of our SCA-TSVAD method for ad-hoc microphone settings. Figure 1 shows the model architecture. Let C, N, D, T and L represent the number of audio channels, speakers, target-speaker embedding dimensions, time frames and log mel-filterbank feature dimensions respectively. For each channel i, we extract the target-speaker embedding $\mathbf{E}_i \in \mathbb{R}^{N\times D}$ corresponding to N speakers using a pre-trained speaker embedding model [7]. These channel-specific embeddings are then aggregated into a $\mathcal{E} = (\mathbf{E}_1, \ldots, \mathbf{E}_i, \ldots, \mathbf{E}_C)$. Then, audio signals are used to compute the log mel-filterbank features $\mathcal{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_i, \ldots, \mathbf{X}_C)$, where $\mathbf{X}_i \in \mathbb{R}^{F\times L}$ is the i-th channel log mel-filterbank of F frames. Next, these features are fed into our front-end model to generate frame-level speaker features $\hat{\mathcal{S}}$. The front-end feature extraction employs a ResNet34 [16] architecture, which processes input L-dim log mel-filterbank feature, followed by segmental statistical pooling (SSP) [17] to compute frame-level representations which are then projected into D-dim speaker features $\hat{\mathcal{S}} = (\hat{\mathbf{S}}_1, \ldots, \hat{\mathbf{S}}_i, \ldots, \hat{\mathbf{S}}_C)$, where $\hat{\mathbf{S}}_i \in \mathbb{R}^{T\times D}$ is the frame-level speaker features from the i-th channel. Now we have frame-level speaker features $\hat{\mathcal{S}} \in \mathbb{R}^{C\times T\times D}$ and target speaker embeddings $\mathcal{E} \in \mathbb{R}^{C\times N\times D}$. Later, the frame-level speaker features are replicated N times

Corresponding Author: Jing Feng, Ming Li

[1]https://github.com/SCATSVAD/DataSimu

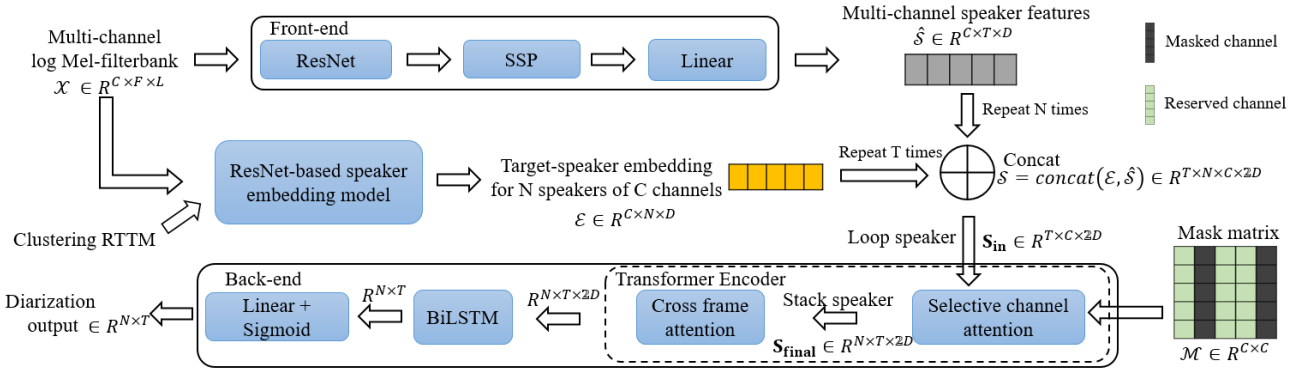

Figure 1: *The architecture of the SCA-TSVAD model. C is the maximum channel number in the dataset, F is the number of frames in the log mel-filterbank, N is the number of speakers, L is the dimension of the log mel-filterbank, T is the number of time frames in speaker features, and D is the dimension of speaker features.*

along the speaker dimension, while the target-speaker embedding is duplicated T times across the temporal axis, followed by concatenation of these two tensors along the embedding dimension:

$$\mathcal{S} = concat(\mathcal{E}, \hat{\mathcal{S}}) \in \mathbb{R}^{T\times N\times C\times 2D} \tag{1}$$

Selective channel attention relies on a mask matrix $\mathcal{M} \in \mathbb{R}^{C\times C}$ based on the C-channel speaker features. For each pair of channels i and j, if $\mathcal{M}_{ij} = 1$, the value will be included in the attention computation for channel i attending to channel j. To block j-th channel (depicted as a black column), we set the j-th column of each row to $-inf$. The columns for active channels (visually represented as green columns in the figure) are set to 1. Additionally, a channel weight matrix $\mathbf{W}_{ch} \in \mathbb{R}^{C\times 1}$ is initialized with ones, where the values corresponding to the masked channels are set to 0, ensuring that only unmasked channels contribute to the output.

Thereafter, to align with the mask matrix dimension, the concatenated embeddings cannot be directly used as input for selective attention, and the concatenated embeddings of each speaker denoted by $\mathbf{S}_{in} \in \mathbb{R}^{T\times C\times 2D}$ must be processed one by one. The selective channel attention takes $\mathbf{S}_{in}$ and the mask matrix, after being replicated T times to $\mathbb{R}^{T\times C\times C}$ as jointly input.

$$\mathbf{Q}^h = \mathbf{W}_Q^h S_{in} + \mathbf{b}_Q^h \tag{2}$$

$$\mathbf{K}^h = \mathbf{W}_K^h S_{in} + \mathbf{b}_K^h \tag{3}$$

$$\mathbf{V}^h = \mathbf{W}_V^h S_{in} + \mathbf{b}_V^h \tag{4}$$

$$\text{Att}(\mathbf{Q}^h, \mathbf{K}^h, \mathbf{V}^h) = \text{softmax}(\frac{\mathbf{Q}^h(\mathbf{K}^h)^\mathbf{T}}{\sqrt{E}} + \mathbf{M})\mathbf{V}^h \tag{5}$$

where $\mathbf{W}^h \in \mathbb{R}^{E\times D}$ is the weight and $\mathbf{b}^h \in \mathbb{R}^E$ is the bias for the h-th head, Att is an abbreviation for attention, $\mathbf{Q}^h$, $\mathbf{K}^h$ and $\mathbf{V}^h$ denotes the query, key and value matrices for the h-th head, $\mathbf{M} \in \mathbb{R}^{T\times C\times C}$ is the extended mask matrix and E represents the dimension of the query vectors. Next, a positional-wise feed-forward layer with a ReLU activation is applied to generate the output, where layer norm and residual connections are employed between each layer. The single speaker output obtained from selective channel attention is represented as $\mathbf{S}_{out} \in \mathbb{R}^{T\times C\times 2D}$ and the output tensor undergoes weighted averaging using the weight matrix.

$$\mathbf{O} = \frac{\sum_{c=1}^{C} \mathbf{S}_{out} \cdot \mathbf{W}_{ch}}{\sum_{c=1}^{C} \mathbf{W}_{ch}} \tag{6}$$

Then, all the $\mathbf{O} \in \mathbb{R}^{T\times 2D}$ representations corresponding to N speakers are stacked together to form the final output $\mathbf{S}_{final} \in \mathbb{R}^{N\times T\times 2D}$. Afterwards, the cross-frame self-attention which allows the model to learn contextual relationships between time frames, improving its ability to distinguish between speakers and handle overlapping speech takes the $\mathbf{S}'$ as input. Following this, a bidirectional LSTM (BiLSTM [18]) is applied by incorporating sequential dependencies. Finally, a linear layer and a sigmoid activation function are used to produce the output of SCA-TSVAD.

2.2. Multi-Channel audio data simulation for varied microphone array setups

To validate the robustness of SCA-TSVAD across diverse microphone configurations, we developed a simulated dataset using SonicSim [19]. Built on the embodied AI simulation platform Habitat-sim [20], SonicSim enables multi-level adjustments at the scene, microphone, and source levels, facilitating the generation of highly diverse synthetic data. The simulation process is outlined as follows:

First, a 3D scene from the Matterport3D [21] dataset is imported into SonicSim to establish the acoustic environment. Sound source positions are randomly determined using Habitat-sim's API, with the number of positions matching the number of speakers in the audio.

Next, noise sources and microphones are randomly placed within a circular region of a 6-meter radius in the horizontal plane (X and Z axes) and a 2-meter range in the vertical axis (Y axis) relative to the sound sources. Microphone configurations are generated randomly, with details provided in Section 3.3.1.

For audio generation, to simulate a more realistic conversational scenario audio, speech segments from multiple speakers in the LibriSpeech database are truncated and re-arranged into a single long audio file based on the start times and durations specified in the RTTM file. This process uses predetermined RTTM files from real-world Ali-Meeting databases but with various simulated microphone configurations applied to generate the data in our experiment. This process is repeated for all speakers in a recording. SonicSim calculates the room impulse responses (RIRs) based on the source audio positions and microphone configurations and then convolves the source audio with the corresponding RIRs.

Finally, the convolved audio signals from all speakers and noise sources are combined to simulate a realistic multi-speaker scenario. This approach ensures the dataset captures a wide range of acoustic conditions.

3. Experimental setup

3.1. Data preparation

To handle varying audio input channels, we align the channel dimensions by padding the audio signals and target-speaker embeddings to the maximum channel number C_{max} in the dataset. For an X-channel input, we create a zero tensor of size C_{max}, generate a random permutation of indices from 0 to $X-1$, and use it to assign the original signals and embeddings into the padded tensor. This method effectively pads the original X-channel input to the C_{max} dimension, with the signal channel positions shuffled according to the random permutation.

3.2. Speaker embedding extraction

TSVAD requires an additional speaker embedding extractor to extract the target speaker's voice features, so we use the ResNet-based pre-trained speaker embedding model from [7]. This model, trained with ArcFace [22] on the CN-Celeb [23] and Ali-Meeting datasets, used a margin of 0.2 and softmax prescaling of 32. During the evaluation, cosine similarity was employed for scoring. Since the Ali-Meeting dataset lacks single-speaker utterances, [7] selected non-overlapping speech segments longer than 2 seconds for training and built a trial set from the evaluation set consisting of 10,692 trials from 25 speakers. The model finally achieved an equal error rate (EER) of 3.199% on the Ali-Meeting trial set and was used to extract target speaker embedding and initialize the front-end ResNet-34 in the SCA-TSVAD system.

3.3. Dataset

3.3.1. Simulated dataset

For our dataset simulation, we utilize the same RTTM files from the train, eval, and test sets of the Ali-Meeting [24] dataset and use the LibriSpeech [15] dataset as the sound source, combined with environmental noise from MUSAN [25] and musical noise from the Free Music Archive (FMA) [26] dataset. Regarding the microphone array configuration, many electronic devices today employ an even number of microphones, with their arrangements being predominantly linear or circular. We randomly select the number of microphones from the set $\{1, 2, 4, 6, 8\}$ for the training set. We randomly choose linear or circular arrays when the configuration is not mono or binaural. As a result, there are eight types of microphone arrays with different microphone spacings, and we randomly select 50 audio samples from the Ali-Meeting train set and apply these eight microphone arrays to obtain a training set of 400 audio samples. For a linear arrangement, the microphone spacing is randomly set within the range of 20 to 40 mm, while for a circular array, the diameter is chosen to be between 82 and 122 mm. Given that the circular array in Ali-Meeting has a diameter of 102 mm, we defined a 20 mm margin on either side.

For the validation set, we fix the linear array spacings at 15 mm, 40 mm, and 65 mm, and the circular array diameters at75 mm, 100 mm, and 130 mm. For the test set, we add two more values for each configuration: linear array spacings of 30 mm and 55 mm, and circular array diameters of 90 mm and 115 mm, to ensure a broader range. Both sets use even numbers of microphones with linear and circular array configurations. Apart from microphone settings, other variables (e.g., room, sound sources, and noise positions) remaine consistent across validation and test sets for each audio instance. The validation set has 16 distinct topologies, making it 16 times larger than the Ali-Meeting eval set. For the test set, we select eight audio samples from the Ali-Meeting test set, with corresponding RTTM files, to simulate the configurations of 27 microphone topologies (as shown in Table 1).

3.3.2. Modified-Ali-Meeting dataset

To validate our SCA-TSVAD model on a real-world dataset, we modify the Ali-Meeting dataset [24] to address the challenge of multi-channel input audio with potentially missing channels due to microphone malfunctions. Specifically, we use the training set of Ali-Meeting, applying the method in [27] to simulate additional data for training—the simulation process follows [7]. During training, we generate a random number less than the maximum number of channels and create corresponding channel permutations, modifying the active channels of the input audio. The modified version of the Ali-Meeting evaluation set is used for validation, and the test set is used for inference, both with various subsets of microphones available.

3.4. Training process

In our experiments, we employ a 2-layer and 2-head Transformer Encoder as the selective channel attention and cross-frame attention layer, and the hidden dimension is 1024. All training audio signals are segmented into 16 s chunks. The model input consists of 80-dimensional log mel-filterbank energies extracted with a 25 ms frame length and a 10 ms frame shift. The dimension of speaker features is 128. The number of speakers is set to 4, and if it is not enough, it will be randomly added from the audio of the same channel. The training is carried out using the binary cross-entropy (BCE) loss function and Adam optimizer. We train our model in three steps. Firstly, the front-end ResNet-34 is frozen, and only the back-end parameters are trained for 20 epochs. Then, all model parameters are unfrozen and trained for another 20 epochs. The first two steps have the same learning rate of $1e$-4. Lastly, our model is fine-tuned with a reduced learning rate of $1e$-5 for 100 epochs to refine the parameters further. The five checkpoints with the lowest validation loss are averaged to obtain the final model for evaluation and inference.

When utilizing the generated dataset with multi-channel and multi-microphone configurations, all three training stages rely entirely on our generated dataset. During training with the Ali-Meeting dataset, the first two stages employ online simulated data, as detailed in Section 3.3.2. In contrast, the third stage utilizes real Ali-Meeting data for training, with a random selection of 1 to 8 channels retained. A specific channel arrangement is selected for each number of retained channels for model validation. The channel arrangements used during testing are presented in Table 2.

3.5. Inference settings

In the inference stage, an AHC-based diarization system [28] initially generates a preliminary result shown in Table 2. Then, non-overlapped speech regions for each speaker are selected to extract target-speaker embeddings using a pre-trained speaker embedding model [7]. Following this, silence regions are removed based on the oracle VAD, and the audio signals are segmented into 16 s chunks with a 4 s shift. After obtaining the probabilities from the SCA-TSVAD model, these results can also serve as the input for the next round of inference. Through multiple iterations in the testing, we observe that the results show minimal improvement beyond the third iteration and, in some cases, even deteriorate. Therefore, we report the results from the second iteration.

4. Experimental results

4.1. Simulated dataset

Table 1 shows the results of our SCA-TSVAD on the simulated test set. Missed speaker time (MI), false alarm speaker time (FA), speaker confusion time (CF), and Diarization Error Rate (DER) are reported. The test set focuses on circular arrays with

Table 1: *DER(%) of different microphone settings on the simulated test set with oracle VAD (collar=250 ms). Mic. Num. means the number of microphones. D. means distance between adjacent linear microphones and diameter for circular array.*

Mic. Num.	Array	D	MI	FA	CF	DER
1	Mono	-	2.7	1.96	1.85	6.50
2	Binaural	-	2.67	1.60	1.52	5.78
4	linear	15	3.64	1.01	0.68	5.33
		30	3.41	1.02	0.78	5.22
		40	3.48	1.04	0.65	**5.17**
		55	3.62	1.08	0.66	5.36
		65	4.02	1.04	0.65	**5.17**
	circular	75	3.32	0.99	0.66	**4.61**
		90	3.25	0.94	0.70	4.89
		100	3.04	0.93	0.63	**4.61**
		115	3.34	0.96	0.57	4.88
		130	3.29	1.00	0.77	5.06
6	linear	15	3.64	0.84	0.64	5.11
		30	3.39	1.04	0.64	5.07
		40	3.78	0.94	0.57	5.29
		55	3.52	0.98	0.55	**5.05**
		65	4.05	0.81	0.50	5.35
	circular	75	3.51	0.99	0.83	5.32
		90	3.53	1.05	0.55	5.13
		100	3.91	0.75	0.58	5.25
		115	3.67	1.05	0.56	5.28
		130	3.45	0.98	0.69	**5.12**
8	circular	75	3.19	0.90	0.55	**4.68**
		90	3.25	1.01	0.70	4.96
		100	3.18	0.95	0.55	**4.68**
		115	3.51	0.74	0.50	4.74
		130	3.34	0.87	0.70	4.91

eight microphones, which are more common and align with Ali-Meeting. Additionally, we include two spacing values per group, the first and last ones unseen during training, evaluating the model's ability to generalize across microphone spacings.

Four out of the five groups achieve the lowest DER under conditions where the microphone spacing deviated from the range used in training, demonstrating that SCA-TSVAD exhibits strong generalization capabilities and can effectively handle microphone configurations not encountered during training. Furthermore, increasing the number of microphones for linear arrays tends to reduce the DER, whereas this effect is not observed with circular arrays.

The comparative analysis in the table suggests that circular arrays generally outperform linear arrays given the same number of microphones in our setup, likely due to their more uniform spatial coverage. Interestingly, the 8-microphone circular array shows slightly higher DER than the 4-microphone configuration, which may result from signal redundancy introduced by the doubled number of microphones.

4.2. Ali-Meeting dataset

The performance on the Ali-Meeting test set is outlined in Table 2. It is evident from the table that our model outperforms most current models in handling all eight channels. This may be because randomly selecting the number of channels during training reduces the model's reliance on specific channel structures and mitigates overfitting to fixed arrangements. Additionally, we test the performance of the SCA-TSVAD model using five different microphone arrangements, ranging from 2 to 7 microphones. The results demonstrate that with the same number of microphones, DER varies only slightly across different channel arrangements. As the channel number increases, the DER gradually declines and eventually leads to stabilization.

Table 2: *DER(%) of different channel arrangements on Ali-Meeting test set with oracle VAD (collar=250 ms). Ch. Num. means the number of channels, Arr. means the arrangements of channel and number 0-7 in the Arr. represents channels 1-8.*

Ch. Num.	Arr.	DER	Model Name/Average
1	0	**4.72**	SCA-TSVAD(our)
	-	6.11*	EEND-M2F+FT [29]
	-	10.20*	WavLM-updated [30]
	-	13.65	AHC-based clustering
2	01	3.76	3.75
	12	3.72	
	23	3.91	
	34	3.68	
	45	3.67	
3	012	3.13	3.16
	123	3.22	
	234	3.11	
	345	3.17	
	456	3.15	
4	0123	2.81	2.76
	1234	2.74	
	2345	2.76	
	3456	2.75	
	4567	2.76	
5	01234	2.46	2.43
	12345	2.46	
	23456	2.39	
	34567	2.51	
	01345	2.33	
6	012345	2.29	2.25
	123456	2.28	
	234567	2.28	
	012456	2.22	
	123567	2.20	
7	0123456	2.19	2.19
	1234567	2.20	
	0134567	2.22	
	0123567	2.16	
	0123457	2.17	
8	-	**2.16**	SCA-TSVAD(our)
	-	2.98	MC-TSVAD [7]
	-	3.98	FFM-TS-VAD [31]
	-	15.67	Official baseline [24]

* These results are obtained without using oracle VAD

5. Conclusion and future work

In this study, we simulated a database featuring diverse microphone configurations and multiple speakers. Additionally, we introduced the SCA-TSVAD model, a unified speaker diarization system capable of handling the dataset of various microphone configurations. Our model demonstrates strong performance on simulated databases. For future work, we plan to evaluate the performance of the SCA-TSVAD model on other multi-channel databases. Furthermore, since the SCA-TSVAD model does not incorporate multi-channel audio's Direction of Arrival (DOA), we aim to further enhance its performance.

6. Acknowledgement

This research is funded in part by the National Natural Science Foundation of China (62171207), Yangtze River Delta Science and Technology Innovation Community Joint Research Project (2024CSJGG01100), Science and Technology Program of Suzhou City(SYC2022051) and Guangdong Science and Technology Plan (2023A1111120012). Many thanks for the computational resource provided by the Advanced Computing East China Sub-Center.

7. References

[1] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," in *Proc. of TASLP*, vol. 14, no. 5, 2006, pp. 1557–1565.

[2] H. Ning, M. Liu, H. Tang, and T. S. Huang, "A spectral clustering approach to speaker diarization." in *Proc. of INTERSPEECH*, 2006.

[3] Q. Lin, R. Yin, M. Li, H. Bredin, and C. Barras, "Lstm based similarity measurement with spectral clustering for speaker diarization," *arXiv preprint arXiv:1907.10393*, 2019.

[4] X. Xiao, N. Kanda, Z. Chen, T. Zhou, T. Yoshioka, S. Chen, Y. Zhao, G. Liu, Y. Wu, J. Wu *et al.*, "Microsoft speaker diarization system for the voxceleb speaker recognition challenge 2020," in *Proc. of ICASSP*, 2021, pp. 5824–5828.

[5] 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. of INTERSPEECH*, 2020, pp. 274–278.

[6] S. Horiguchi, P. Garcia, Y. Fujita, S. Watanabe, and K. Nagamatsu, "End-to-end speaker diarization as post-processing," in *Proc. of ICASSP*, 2021, pp. 7188–7192.

[7] W. Wang, X. Qin, and M. Li, "Cross-channel attention-based target speaker voice activity detection: Experimental results for the m2met challenge," in *Proc. of ICASSP*, 2022, pp. 9171–9175.

[8] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," in *Proc. of TASLP*, vol. 20, no. 2, 2012, pp. 356–370.

[9] M. T. Ho, J. Lee, B.-K. Lee, D. H. Yi, and H.-G. Kang, "A cross-channel attention-based wave-u-net for multi-channel speech enhancement." in *Proc. of INTERSPEECH*, 2020, pp. 4049–4053.

[10] D. Wang, Z. Chen, and T. Yoshioka, "Neural speech separation using spatially distributed microphones," in *Proc. of INTERSPEECH*, 2020, pp. 339–343.

[11] D. Wang, T. Yoshioka, Z. Chen, X. Wang, T. Zhou, and Z. Meng, "Continuous speech separation with ad hoc microphone arrays," in *Proc. of EUSIPCO*, 2021, pp. 1100–1104.

[12] F.-J. Chang, M. Radfar, A. Mouchtaris, and M. Omologo, "Multi-channel transformer transducer for speech recognition," in *Proc. of INTERSPEECH*, 2021, pp. 296–300.

[13] S. Horiguchi, Y. Takashima, P. Garcia, S. Watanabe, and Y. Kawaguchi, "Multi-channel end-to-end neural diarization with distributed microphones," in *Proc. of ICASSP*, 2022, pp. 7332–7336.

[14] C. Liang, J. Chen, S. Guan, and X.-L. Zhang, "Attention-based multi-channel speaker verification with ad-hoc microphone arrays," in *Proc. of APSIPA ASC*, 2021, pp. 1111–1115.

[15] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Proc. of ICASSP*, 2015, pp. 5206–5210.

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

[17] W. Wang, Q. Lin, D. Cai, and M. Li, "Similarity measurement of segment-level speaker embeddings in speaker diarization," in *Proc. of TASLP*, vol. 30. IEEE, 2022, pp. 2645–2658.

[18] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of lstm and bilstm in forecasting time series," in *Proc. of Big Data*, 2019, pp. 3285–3292.

[19] K. Li, W. Sang, C. Zeng, R. Yang, G. Chen, and X. Hu, "Sonicsim: A customizable simulation platform for speech processing in moving sound source scenarios," in *Proc. of ICLR*, 2025.

[20] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik *et al.*, "Habitat: A platform for embodied ai research," in *Proc. of ICCV*, 2019, pp. 9339–9347.

[21] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," in *Proc. of 3DV*, 2017, pp. 667–676.

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

[23] Y. Fan, J. Kang, L. Li, K. Li, H. Chen, S. Cheng, P. Zhang, Z. Zhou, Y. Cai, and D. Wang, "Cn-celeb: a challenging chinese speaker recognition dataset," in *Proc. of ICASSP*, 2020, pp. 7604–7608.

[24] F. Yu, S. Zhang, Y. Fu, L. Xie, S. Zheng, Z. Du, W. Huang, P. Guo, Z. Yan, B. Ma *et al.*, "M2met: The icassp 2022 multi-channel multi-party meeting transcription challenge," in *Proc. of ICASSP*, 2022, pp. 6167–6171.

[25] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[26] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, "Fma: A dataset for music analysis," in *Proc. of ISMIR*, 2016.

[27] W. Wang, D. Cai, Q. Lin, L. Yang, J. Wang, J. Wang, and M. Li, "The dku-dukeece-lenovo system for the diarization task of the 2021 voxceleb speaker recognition challenge," *arXiv preprint arXiv:2109.02002*, 2021.

[28] 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. of INTERSPEECH*, 2018, pp. 2808–2812.

[29] M. Härkönen, S. J. Broughton, and L. Samarakoon, "Eend-m2f: Masked-attention mask transformers for speaker diarization," in *Proc. of INTERSPEECH*, 2024, pp. 37–41.

[30] J. Han, F. Landini, J. Rohdin, A. Silnova, M. Diez, and L. Burget, "Leveraging self-supervised learning for speaker diarization," *arXiv preprint arXiv:2409.09408*, 2024.

[31] N. Zheng, N. Li, X. Wu, L. Meng, J. Kang, H. Wu, C. Weng, D. Su, and H. Meng, "The cuhk-tencent speaker diarization system for the icassp 2022 multi-channel multi-party meeting transcription challenge," in *Proc. of ICASSP*, 2022, pp. 9161–9165.