Interspeech 2025
17-21 August 2025, Rotterdam, The Netherlands

Towards Robust Overlapping Speech Detection: A Speaker-Aware Progressive Approach Using WavLM

Zhaokai Sun[1,2], *Li Zhang*[1], *Qing Wang*[1], *Pan Zhou*[2], *Lei Xie*[1*]

[1]Audio, Speech and Language Processing Group (ASLP@NPU), School of Software, Northwestern Polytechnical University, China
[2]Space AI, Li Auto

zksun@mail.nwpu.edu.cn, lxie@nwpu.edu.cn

Abstract

Overlapping Speech Detection (OSD) aims to identify regions where multiple speakers overlap in a conversation, a critical challenge in multi-party speech processing. This work proposes a speaker-aware progressive OSD model that leverages a progressive training strategy to enhance the correlation between subtasks such as voice activity detection (VAD) and overlap detection. To improve acoustic representation, we explore the effectiveness of state-of-the-art self-supervised learning (SSL) models, including WavLM and wav2vec 2.0, while incorporating a speaker attention module to enrich features with frame-level speaker information. Experimental results show that the proposed method achieves state-of-the-art performance, with an F1 score of 82.76% on the AMI test set, demonstrating its robustness and effectiveness in OSD.

Index Terms: overlapped speech detection, speaker recognition, multi-task learning

1. Introduction

Multi-party conversation recognition is one of the most challenging problems in speech processing [1], particularly due to difficulties in accurately identifying overlapping speech regions. Overlapping regions present unique challenges, as they involve both overlap of speech segments and speaker identities. Effectively addressing these challenges is critical for applications such as speaker diarization, automatic speech recognition (ASR), and multi-speaker dialogue systems [2, 3, 4]. Overlap Speech Detection (OSD) has emerged as a crucial component in tackling these challenges by identifying overlapping speech regions in complex acoustic environments.

Recent advancements in deep learning have introduced powerful tools for OSD, leveraging convolutional neural networks (CNNs), long short-term memory (LSTM) networks, and transformer architectures [5, 6, 7]. These methods process diverse input features such as raw waveforms, filter banks (FBank), spectrograms, and x-vectors [8, 9, 10, 11]. However, the robustness of these systems remains closely tied to the diversity and quality of training data. Due to the scarcity of real-world dialogue datasets with sufficient overlap durations, most OSD research relies on simulated data or selectively filtered open-source datasets. Simulated datasets, such as LibriSpeech-derived LibriHeavyMix [12], are generated based on statistical characteristics of overlapping and non-overlapping regions, but often exhibit significant performance variability across test conditions due to domain mismatches. In [13], model trained on simulated datasets shows performance degradation on relastic test datasets. Open-source datasets, including AliMeeting [14] and AMI [15], also serve as benchmarks. In [9], Yin et al. propose a large-scale benchmark, with filtered open-source datasets.

Multi-task learning (MTL) [16] has shown promise in improving OSD performance by integrating related tasks, such as voice activity detection (VAD). Incorporating VAD as a subtask helps capture human speech patterns and enhances the overall robustness of OSD systems. However, many existing approaches treat MTL as a three-class classification problem, which often underutilizes the intrinsic relationships between tasks [17]. For instance, models like CRNN [18] classify audio frames into silence, single-speaker, and overlap categories, but this simplification can lead to imbalances in precision and recall.

In addition to temporal overlap detection, distinguishing speaker identities within overlapping regions is another critical challenge. Speaker recognition, another critical component of speech analysis, provides valuable insights for addressing overlapping speech [19]. By extracting speaker-specific features, speaker recognition systems enable applications such as access control, personalized voice services, and dialogue analysis. With the development of deep learning, the robustness of speaker recognition systems has gradually improved, as seen in systems like ecapa [20] and campplus [21]. Pretrained self-supervised learning (SSL) models, such as WavLM [22], have demonstrated remarkable effectiveness in capturing speaker information, particularly for overlapping audio scenarios. However, most OSD systems fail to explicitly incorporate frame-level speaker representations, missing an opportunity to improve overlap detection.

To address these challenges, we propose a robust and progressive OSD framework that leverages WavLM and incorporates speaker representations to enhance overlap detection. Our approach introduces a progressive learning strategy to better utilize task interrelationships, enabling stepwise refinement of VAD and OSD components. Specifically, we mask encoder hidden states using VAD logits and feed the masked representations into the OSD decoder. Additionally, inspired by studies highlighting the feature distribution similarities between single-speaker and overlapping segments [23], we design a frame-level speaker attention module. This module extracts speaker representations and integrates them with audio features from the pretrained SSL model, enriching the OSD decoder's input.

To train our system, we utilize a combination of simulated and real-world datasets, including LibriHeavyMix, AliMeeting, and AMI. Our training strategy involves pretraining on simulated datasets followed by fine-tuning on realistic data. This two-step process, commonly used in speaker diarization, facilitates model convergence and enhances robustness across diverse scenarios [24].

The remainder of this paper is structured as follows: Section 2 details the proposed framework and OSD system config-

10.21437/Interspeech.2025-1778

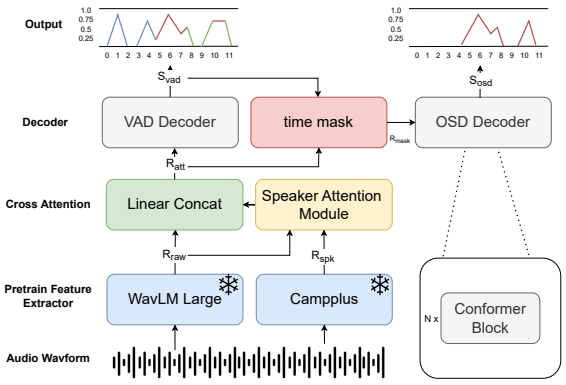


Figure 1: *Our detailed progressive OSD architecture.*

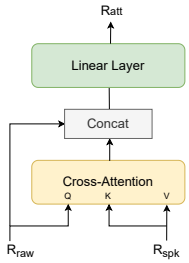


Figure 2: *Frame-level speaker attention module.*

uration. Section 3 presents the experimental setup and results. Section 4 concludes the paper and discusses future directions.

2. Methods

In this section, we provide a comprehensive description of the proposed speaker-aware progressive OSD framework. We begin with an overview of the progressive OSD architecture, followed by a detailed explanation of the progressive OSD training strategy. Finally, we present a thorough discussion of the individual modules within the system.

2.1. Overview

As illustrated in Fig. 1, the proposed speaker-aware progressive OSD model comprises four key components: the pre-trained SSL encoder, the speaker attention module, the temporal masking module, and the decoder. During the inference process, the input audio is combined with speaker information to enrich SSL representations. These fused representations are then masked using the VAD logits before being passed through the OSD decoder.

In this paper, we leverage the representations generated by WavLM-Large[1] to enhance OSD performance. The pre-trained WavLM model offers robust, high-level acoustic representations, as it is trained on tasks such as masked speech prediction and audio denoising [22]. To further refine these representations, we incorporate a speaker attention module, which enriches the WavLM hidden states with speaker-specific information. Additionally, we utilize the pre-trained CampPlus[2] model to generate frame-level speaker embeddings. These fused representations are then passed to the VAD decoder, which consists of Conformer blocks.

For the progressive learning strategy, we design the OSD input of our end-to-end system based on temporal masking, enabling the OSD decoder to focus on relevant speech segments. Specifically, the temporal mask module generates masked fusion representations by utilizing the output logits from the VAD decoder. Finally, the OSD decoder produces the OSD logits using the masked fusion representations.

[1]https://github.com/microsoft/unilm/tree/master/wavlm

[2]https://www.modelscope.cn/models/iic/speech_campplus_sv_zh-cn_16k-common

2.2. Progressive OSD modeling

The progressive modeling approach we propose is depicted in Fig. 1. Previous studies typically generate VAD and OSD predictions simultaneously in the final step, applying classification losses for both tasks [9, 13, 25]. However, these approaches do not explicitly model the relationship between the two tasks. Serving as a speech type classification task, they share significant similarities. VAD, as a fundamental task, ensures high recall for overlapping speech detection, while the OSD model focuses on enhancing detection precision. Our goal is to improve OSD performance by explicitly modeling the relationship between VAD and OSD tasks.

In our approach, we propose a progressive strategy for OSD tasks, where the input to the OSD decoder is influenced by the VAD output. Specifically, we apply a temporal mask to the frame-level acoustic representations from the pre-trained SSL encoder using VAD logits. The temporal mask module effectively masks silent segments, enabling the OSD decoder to focus solely on speech regions, thus enhancing the model's ability to accurately distinguish overlapping speech. The time masking process is formalized in Eq 8, where R_{att} represents the hidden features enhanced by the speaker attention module, and S_{vad} denotes the VAD logits.

2.3. Pretrain SSL Encoder

WavLM is a self-supervised speech processing model designed to leverage diverse speech datasets. It adopts an encoder-only Transformer architecture, where the encoder consists of seven convolutional layers followed by a stack of Transformer blocks with gated relative position bias [22]. Trained on an audio mask prediction task, WavLM has demonstrated exceptional performance across various speech processing tasks, including speaker recognition, speech recognition, and speech generation. Moreover, recent studies have explored its robustness in additional downstream applications, such as speech event detection [26] and spoofed audio detection [27]. In this paper, we adapt WavLM for OSD tasks. As illustrated in Fig. 1, the WavLM encoder converts input audio signals into a sequence of high-level abstract representations, serving as the foundation for subsequent processing.

The input audio waveform, denoted as X, with corresponding VAD, OSD labels given by Y_{vad} and Y_{osd}. The WavLM encoder is F_w. After T length audio waveform resampled to 16kHz, we extract the encoder audio representations R_{raw} from pre-trained encoder as follows:

$$X = [x_1, x_2, ..., x_t],\ x_t \in \mathbb{R}, t \in [0, T] \tag{1}$$

$$Y_{vad} = [y_{vad_1}, y_{vad_2}, ..., y_{vad_t}],\ y_{vad_t} \in \mathbb{R} \quad (2)$$

$$Y_{osd} = [y_{osd_1}, y_{osd_2}, ..., y_{osd_t}],\ y_{osd_t} \in \mathbb{R} \quad (3)$$

$$R_{raw} = F_w(X) \quad (4)$$

2.4. Speaker Attention Module

Previous research on overlapping speech in diarization tasks has shown that speaker representations of overlap frames exhibit correlations between overlapping speakers [23]. Motivated by this, we incorporate speaker representations into our system to enhance OSD performance. Specifically, we utilize CampPlus, a pretrained speaker verification (SV) model based on the D-TDNN architecture, which has achieved state-of-the-art performance on SV benchmark datasets [21]. Since WavLM is not explicitly optimized for SV tasks, we introduce a speaker attention module to refine WavLM representations by integrating frame-level speaker characteristics extracted from CampPlus.

The Fbank process is denoted as F_{mel}. Given an input audio waveform X, the computation of the Fbank features X_{fbank} follows the procedure described in 5. The CampPlus model, F_{sv}, transforms the Fbank features X_{fbank} into frame-level speaker representations R_{spk}. Subsequently, the cross-attention module F_{att} integrates these speaker representations to enhance the pretrained features R_{raw}. Within the cross-attention module, R_{raw} serves as the query vector, while R_{spk} is used as both the key and value vector..

$$X_{fbank} = F_{mel}(X) \quad (5)$$

$$R_{spk} = F_{sv}(X_{fbank}) \quad (6)$$

$$R_{att} = F_{att}(R_{raw}, R_{spk}, R_{spk}) + R_{raw} \quad (7)$$

2.5. Temporal Masking Module

VAD and OSD tasks exhibit a hierarchical relationship, where OSD can leverage the single-speaker segments identified by VAD to focus on learning overlapping speech features. Unlike conventional multi-task learning, which produces a unified three-class prediction at the backend, our method explicitly models the interaction between VAD and OSD through a temporal masking module. In this module, a soft mask is applied to the fused audio representations obtained in earlier stages, based on the logits generated by the VAD module. The OSD decoder then computes the OSD scores for each frame using these masked representations.

The VAD logits are donted as S_{vad}. The temporal masking module, F_{mask} , takes the fused audio representation R_{att} along with the VAD logits to produce the masked representation R_{mask}. This process is formally defined by the following equation:

$$R_{mask} = F_{mask}(S_{vad},\ R_{att}) \quad (8)$$

2.6. Decoder

The decoder module transforms the frame-level features extracted in previous stages into prediction scores. In OSD tasks, the decoder is typically composed of multiple linear layers, LSTM networks, or Transformer-based architectures. In our system, we implement the decoder using several Conformer blocks. Given that the OSD task is trained using a progressive learning strategy, separate decoders are employed for VAD and OSD. Both decoders share the same network architecture within our framework.

The VAD decoder F_{vad} utilizes the attention-enhanced representation R_{att} to produce VAD logits S_{vad} whereas the OSD decoder F_{osd} operates on the masked representation R_{mask} to generate OSD logits S_{osd}.

$$S_{vad} = F_{vad}(R_{att}) \quad (9)$$

$$S_{osd} = F_{osd}(R_{mask}) \quad (10)$$

To address the uncertainty introduced by human annotations, we employ a fuzzy labeling strategy. In this approach, overlapping speech frames are assigned a label of 1, with a linear decay applied near the boundaries to reflect the gradual transition between speech regions. Given this label design, our two-stage model is trained as a regression task using mean squared error (MSE) loss. Prior research [13] has demonstrated the effectiveness of fuzzy labeling in OSD tasks. The frame-level labels are denoted as y, and the loss functions for both tasks are formulated as follows.

$$L_{vad} = \frac{1}{T}\sum_{i=1}^{T} ||y_{vad_i} - S_{vad_i}||,\ S_{vad_i} \in S_{vad} \quad (11)$$

$$L_{osd} = \frac{1}{T}\sum_{i=1}^{T} ||y_{osd_i} - S_{osd_i}||,\ S_{osd_i} \in S_{osd} \quad (12)$$

3. Experiments

3.1. Experiments setup

In this section, we describe the experimental setup used for our paper, including the datasets and training configurations.

3.1.1. Data Corpus

An ideal OSD system should be robust to variations in language, speaking style, and speaker-to-microphone distance. To enhance generalization, we compile a diverse multi-speaker conversational dataset from various open-source sources. During training, the model is first pretrained for five epochs on Libri-HeavyMix [12] before being fine-tuned using realistic data. The details of the datasets are presented in Table 1.

One major challenge in OSD training is the inherent class imbalance, as overlapping speech occurs far less frequently than non-overlapping speech. To mitigate this issue, we manually curate the training data to ensure a balanced distribution, maintaining a 1:1:1 ratio among silent segments, single-speaker segments, and overlapping speech segments within each session.

Furthermore, to account for annotation uncertainty, we apply a fuzzy labeling strategy near VAD and OSD boundaries. Specifically, the overlapping speech labels are gradually decayed from 1 to 0 over a span of 10 frames near the boundary, facilitating a smoother transition and improving model robustness.

3.1.2. Training Configuration

The speech data is segmented into 5-second intervals. To ensure compatibility with the pretrained frontend, the audio frame length is set to 25 ms with a frame shift of 20 ms. All experimental setups maintain consistent batch sizes and optimizer configurations. Specifically, we employ the Adam optimizer with an initial learning rate of 1e-4 for training all OSD models. To prevent the model from converging to suboptimal solutions due to a fixed learning rate, we apply a weight decay of 1e-4 and utilize a cosine scheduler to dynamically adjust the learning rate, facilitating a smoother optimization process.

Table 1: Details of datasets. LibriheavyMix [12] is an open-source simulation conversastion dataset.

Dataset	Style	Hours	Overlap Ratio (%)
Alimeeting	Meeting/Far	104.75	42.27
AMI	Meeting/Far	75	19
LibriheavyMix	Multi/Near	240	42.56

Table 2: Results(%) of multi-task model on AMI test.

Method	Recall	Precision	F1
CNN [2]	44.6	75.8	56.1
x-vectors [15]	47.7	85.5	61.0
pyannote [16]	80.7	70.5	75.3
conformer [17]	65.03	64.43	64.73
xlsr-conformer [18]	79.38	79.04	79.21
ours (AMI)	**81.48**	**84.08**	**82.76**
ours (Alimeeting)	**79.79**	**83.25**	**81.48**

Additionally, we investigate the impact of reverberation augmentation during training. However, our experiments indicate that adding reverberation does not yield significant performance improvements; therefore, it is not incorporated into our final training pipeline.

3.2. Experiment Results

To comprehensively investigate the effectiveness of our approach, we conduct four main experiments. Table 2 provides a comparison between our system and the most competitive existing methods. Table 3 examines the impact of the speaker attention module. Table 4 evaluates the improvements introduced by the progressive learning strategy across different system configurations. Finally, Table 5 explores the effect of different pretrained SSL encoders on feature extraction by substituting alternative models.

The first experiment aims to evaluate the performance of the proposed progressive OSD system in comparison with related studies. As shown in Table 2, our system achieves an F1 score of 82.76%, outperforming all previously reported methods. Notably, it demonstrates a relative improvement of 4.4% over the previous state-of-the-art system, XLSR-Conformer. Additionally, both recall and precision scores show significant enhancements, indicating that the progressive learning approach effectively boosts overall system performance.

The second experiment examines different methods for integrating speaker information into the OSD system. In our proposed framework, speaker information is incorporated through the speaker attention module. To provide a comparative analysis, we also implemented a previously proposed approach that aligns OSD representations with speaker features using mean squared error (MSE) loss [23]. As presented in Table 3, the progressive OSD system with the speaker attention module achieves an F1 score of 82.76% on the AMI test set, while the system using MSE attains 81.62%. These findings suggest that, compared to MSE-based alignment, the attention mechanism enables more effective utilization of speaker information, leading to improved system performance.

Table 3: Results(%) of speaker ablation experiments.

Method	Recall	Precision	F1
p-OSD-wavlm-spkAtt (ours)	**81.48**	**84.08**	**82.76**
p-OSD-wavlm-spkMSE	80.80	82.45	81.62
p-OSD-wavlm	79.43	79.52	79.47

Table 4: Results(%) of progressive ablation experiments, where p/u means progressive or unified approach.

Method	Recall	Precision	F1
p-OSD-wavlm-spkAtt (ours)	**81.48**	**84.08**	**82.76**
u-OSD-wavlm-spkAtt	80.90	83.55	82.20
p-OSD-wavlm-spkMSE	80.80	82.45	81.62
u-OSD-wavlm-spkMSE	80.52	81.92	81.21
p-OSD	66.82	65.10	65.95
u-OSD	65.03	64.43	64.73

Table 5: Results(%) of SSL encoder ablation experiments.

Method	Recall	Precision	F1
p-OSD-wavlm-spkAtt (ours)	**81.48**	**84.08**	**82.76**
p-OSD-xlsr-spkAtt	79.19	80.97	80.07
p-OSD-wavlm	79.43	79.52	79.47
p-OSD-xlsr	78.45	79.03	78.73
P-OSD	66.82	65.10	65.95

The third experiment aims to demonstrate the effectiveness of our two-stage approach in improving recall and reducing the false alarm rate in the OSD task. To ensure a fair comparison, we modified the multi-task modeling strategy used in previous OSD systems into a progressive approach while maintaining an identical modeling framework. As shown in Table 4, our progressive OSD system, incorporating WavLM and the speaker attention module, achieves the highest F1 score, with a relative improvement of 0.6%. The comparison between the unified and progressive approaches indicates that the progressive model yields superior recall and false alarm rates, further validating the effectiveness of the proposed progressive learning strategy.

4. Conclusion

In this paper, we propose a speaker-aware progressive OSD model that leverages a progressive learning strategy to effectively exploit the correlations between subtasks, thereby enhancing the robustness and performance of OSD systems. Experimental results consistently demonstrate the advantages of this approach across different configurations. By integrating a pretrained SSL frontend with a speaker attention module, the system achieves enriched acoustic representations, further boosting detection accuracy. The proposed speaker-aware progressive OSD model achieves superior performance compared to previous state-of-the-art methods, underscoring the effectiveness of the progressive learning strategy and the speaker-enhanced acoustic feature design.

5. References

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

[2] K. Boakye, B. Trueba-Hornero, O. Vinyals, and G. Friedland, "Overlapped speech detection for improved speaker diarization in multiparty meetings," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 4353–4356.

[3] D. Charlet, C. Barras, and J.-S. Liénard, "Impact of overlapping speech detection on speaker diarization for broadcast news and debates," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7707–7711.

[4] M. Diez, F. Landini, L. Burget, J. Rohdin, A. Silnova, K. Žmolíková, O. Novotný, K. Veselý, O. Glembek, O. Plchot, L. Mošner, and P. Matějka, "But system for dihard speech diarization challenge 2018," in *Interspeech 2018*, 2018, pp. 2798–2802.

[5] M. Kunešová, M. Hrúz, Z. Zajíc, and V. Radová, "Detection of overlapping speech for the purposes of speaker diarization," in *Speech and Computer*, A. A. Salah, A. Karpov, and R. Potapova, Eds. Cham: Springer International Publishing, 2019, pp. 247–257.

[6] L. Bullock, H. Bredin, and L. P. Garcia-Perera, "Overlap-aware diarization: resegmentation using neural end-to-end overlapped speech detection," 2019. [Online]. Available: https://arxiv.org/abs/1910.11646

[7] J. Wang, Y. Liu, B. Wang, Y. Zhi, S. Li, S. Xia, J. Zhang, F. Tong, L. Li, and Q. Hong, "Spatial-aware speaker diarization for multi-channel multi-party meeting," *arXiv preprint arXiv:2209.12002*, 2022.

[8] H. Bredin and A. Laurent, "End-to-end speaker segmentation for overlap-aware resegmentation," 2021. [Online]. Available: https://arxiv.org/abs/2104.04045

[9] Z. Yin, J. Tian, X. Hu, X. Xu, and Y. Xiang, "Large-scale learning on overlapped speech detection: New benchmark and new general system," 2023. [Online]. Available: https://arxiv.org/abs/2308.05987

[10] E. Kazimirova and A. Belyaev, "Automatic detection of multi-speaker fragments with high time resolution," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2018, pp. 1388–1392.

[11] L. Mateju, F. Kynych, P. Cerva, J. Malek, and J. Zdánskỳ, "Overlapped speech detection in broadcast streams using x-vectors." in *INTERSPEECH*, 2022, pp. 4606–4610.

[12] Z. Jin, Y. Yang, M. Shi, W. Kang, X. Yang, Z. Yao, F. Kuang, L. Guo, L. Meng, L. Lin *et al.*, "Libriheavymix: a 20,000-hour dataset for single-channel reverberant multi-talker speech separation, asr and speaker diarization," *arXiv preprint arXiv:2409.00819*, 2024.

[13] M. Kunešová and Z. Zajíc, "Multitask detection of speaker changes, overlapping speech and voice activity using wav2vec 2.0," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[14] 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 *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 6167–6171.

[15] S. Renals, T. Hain, and H. Bourlard, "Recognition and understanding of meetings the ami and amida projects," in *2007 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, 2007, pp. 238–247.

[16] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE transactions on knowledge and data engineering*, vol. 34, no. 12, pp. 5586–5609, 2021.

[17] J.-w. Jung, H.-S. Heo, Y. Kwon, J. S. Chung, and B.-J. Lee, "Three-class overlapped speech detection using a convolutional recurrent neural network," *arXiv preprint arXiv:2104.02878*, 2021.

[18] S. Zheng, S. Zhang, W. Huang, Q. Chen, H. Suo, M. Lei, J. Feng, and Z. Yan, "Beamtransformer: Microphone array-based overlapping speech detection," *arXiv preprint arXiv:2109.04049*, 2021.

[19] Y. Tu, W. Lin, and M.-W. Mak, "A survey on text-dependent and text-independent speaker verification," *IEEE Access*, vol. 10, pp. 99 038–99 049, 2022.

[20] N. Dawalatabad, M. Ravanelli, F. Grondin, J. Thienpondt, B. Desplanques, and H. Na, "Ecapa-tdnn embeddings for speaker diarization," *arXiv preprint arXiv:2104.01466*, 2021.

[21] H. Wang, S. Zheng, Y. Chen, L. Cheng, and Q. Chen, "Cam++: A fast and efficient network for speaker verification using context-aware masking," *arXiv preprint arXiv:2303.00332*, 2023.

[22] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[23] T. Cord-Landwehr, C. Boeddeker, C. Zorilă, R. Doddipatla, and R. Haeb-Umbach, "Frame-wise and overlap-robust speaker embeddings for meeting diarization," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[24] 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*. ISCA, Oct. 2020.

[25] M. Lebourdais, T. Mariotte, M. Tahon, A. Larcher, A. Laurent, S. Montresor, S. Meignier, and J.-H. Thomas, "Joint speech and overlap detection: a benchmark over multiple audio setup and speech domains," *arXiv preprint arXiv:2307.13012*, 2023.

[26] Y. Jiang, R. Tao, W. Huang, Q. Chen, and W. Wang, "Unified audio event detection," 2024. [Online]. Available: https://arxiv.org/abs/2409.08552

[27] Y. Yang, H. Qin, H. Zhou, C. Wang, T. Guo, K. Han, and Y. Wang, "A robust audio deepfake detection system via multi-view feature," 2024. [Online]. Available: https://arxiv.org/abs/2403.01960