



# Speaker Targeting via Self-Speaker Adaptation for Multi-talker ASR

Weiying Wang<sup>1</sup>, Taejin Park<sup>1</sup>, Ivan Medennikov<sup>1</sup>, Jinhao Wang<sup>1</sup>, Kunal Dhawan<sup>1</sup>, He Huang<sup>1</sup>,  
Nithin Rao Koluguri<sup>1</sup>, Jagadeesh Balam<sup>1</sup>, Boris Ginsburg<sup>1</sup>

<sup>1</sup>NVIDIA, USA

{weiqingw,taejinp,imedennikov,jinhanw,kdhawan,heh,nkoluguri,jbalam,bginsburg}@nvidia.com

## Abstract

We propose a self-speaker adaptation method for streaming multi-talker automatic speech recognition (ASR) that eliminates the need for explicit speaker queries. Unlike conventional approaches requiring target speaker embeddings or enrollment audio, our technique dynamically adapts individual ASR instances through speaker-wise speech activity prediction. The key innovation involves injecting speaker-specific kernels generated via speaker supervision activations into selected ASR encoder layers. This enables instantaneous speaker adaptation to target speakers while handling fully overlapped speech even in a streaming scenario. Experiments show state-of-the-art performance in both offline and streaming scenarios, demonstrating that our self-adaptive method effectively addresses severe speech overlap through streamlined speaker-focused recognition. The results validate the proposed self-speaker adaptation approach as a robust solution for multi-talker ASR under severe overlapping speech conditions.

**Index Terms:** Multi-talker ASR, Multi-speaker ASR, Target-speaker ASR, Streaming ASR

## 1. Introduction

Recent advancements in Automatic Speech Recognition (ASR), driven by improved architectures and larger training datasets, have significantly advanced the field. Concurrently, interest in multi-talker ASR has grown, particularly for applications such as analyzing natural conversations, developing voice assistants, and transcribing speech in health and legal contexts. Although this task is referred to by various terms—such as multi-talker ASR, multi-speaker ASR, or sometimes speaker-attributed ASR—the core challenge remains the same. Regardless of the terminology, transcribing speech signals in the presence of overlapping speech from multiple speakers is a demanding task, as ASR systems need to handle significantly increased variability.

Since the early days of ASR research, one of the most significant challenges for ASR systems has been modeling intrinsic and extrinsic variability in speech, often caused by speaker-specific factors such as accent, age, or gender. To tackle these challenges, speaker adaptation techniques were developed to address these variations. These techniques include the use of auxiliary speaker embeddings [1], such as i-vectors [2], which represent speaker-specific traits and are integrated as additional features. Other methods involved feature transformation techniques, such as feature-space maximum likelihood linear regression (fMLLR) [3] and vocal tract length normalization (VTLN) [4], which aim to generate speaker-independent features. Additionally, model-based adaptation techniques, such as learning hidden unit contributions (LHUC) [5] or linear trans-

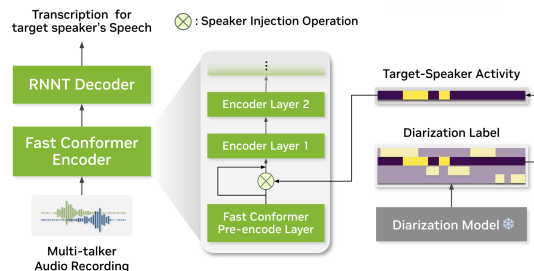


Figure 1: *Speaker injection at pre-encode layer of Fast Conformer encoder.*

forms applied to neural network layers [6, 7], were employed to capture speaker variability within the acoustic modeling framework. More recently, speaker adaptation techniques have been extended to end-to-end ASR models [8, 9].

In multi-talker scenarios, handling frequent overlapped speech poses a significant challenge. To address this, speaker diarization systems are commonly employed to detect and separate the speech of individual speakers. The separated segments are then passed to a single-speaker ASR model for transcription [10]. Techniques such as guided source separation (GSS) are often used to estimate spectral masks [11], facilitating the separation process. A major breakthrough in end-to-end multi-speaker ASR was achieved with the introduction of Serialized Output Training (SOT) in [12]. Unlike earlier multi-talker ASR systems [13, 14], which relied on multiple encoders or heads, SOT serializes overlapped speech, enabling the multi-head self-attention mechanism [15] to align speech signals with their corresponding tokens effectively. Subsequent advancements have led to numerous improved versions of the SOT approach [16, 17, 18].

In parallel, speaker-attributed ASR [19] has significantly improved ASR performance for overlapped multi-talker speech. Recently, alignment-free training (AFT) [20] has shown that high performance can be achieved without explicit alignment by training and predicting each speaker's speech separately. These advancements mark substantial progress in multi-talker ASR.

In this paper, we propose a self-speaker adaptation (SSA) technique that can be effectively repurposed for multi-talker automatic speech recognition (ASR). A common challenge in single speaker streaming ASR systems is their tendency to prioritize a specific speaker—often the one closest to the microphone or the first to appear—while disregarding other speakers in multi-talker scenarios. This behavior arises from the encoder states being optimized to maximize accuracy for a particular speaker, which poses a significant challenge when fine-tuning a single-speaker ASR model for multi-talker applications. Specifically, the fine-tuned model must undergo sub-

stantial weight adjustments to counteract this inherent bias and achieve balanced recognition across all speakers.

On the contrary, we leverage this mechanism in reverse to reinforce the propensity of adhering to a specific speaker. Our proposed technique achieves this by injecting a learnable speaker kernel into the pre-encode layer of the ASR encoder [21]. This enables the ASR encoder to detect speech presence and utilize the speech kernel to adapt the encoder states dynamically. As a result, the encoder becomes more responsive to the targeted speaker’s speech characteristics. However, this approach requires deploying one model instance per speaker, meaning the number of model instances must match the number of speakers. While this necessitates additional computational resources, it significantly enhances multi-talker ASR performance, achieving state-of-the-art error rates on benchmark datasets for multi-talker ASR.

Unlike traditional speaker adaptation methods that rely on external speaker representations, such as i-vectors [2, 1] or neural speaker embeddings [19], our technique depends solely on the speech activity of a specific speaker. For this reason, we refer to our method as self-speaker adaptation. In the experimental section, we compare our proposed method with baseline single-speaker ASR models and other studies reported on the same benchmark datasets, evaluating performance in both of-line and streaming multi-talker ASR scenarios.

## 2. Proposed Method

Typically, target-speaker ASR systems rely on target speaker embeddings or enrollment audio to extract and utilize speaker-specific information from multi-talker utterances. However, the performance of such systems is highly dependent on the quality of the provided queries. For instance, a clean and noise-free query audio is generally preferred to achieve optimal results in target-speaker ASR tasks. In this work, we introduce a novel SSA approach that enables the model to adapt to a specific speaker using only the corresponding speech activities, eliminating the need for high-quality external queries. This method leverages the inherent speaker characteristics present in the input audio, allowing the model to dynamically adjust its focus to better recognize the target speaker’s speech. By doing so, our approach reduces the dependency on external resources and enhances the robustness of the system in real-world scenarios where high-quality queries or speaker representations may not always be available.

### 2.1. Self-Speaker Adaptation

As illustrated in Figure 1, our proposed method incorporates a speaker injection module into one of the layers of a single-speaker ASR model. Specifically, the speech activity is treated as a mask and applied to the output of the selected layer, with a residual connection added to preserve the original information. This process can be formally expressed as:

$$\mathbf{X}_{\text{inj}}^i = f_{\text{inj}}(\mathbf{X}^i, \mathbf{y}_{\text{spk}_k}) + \mathbf{X}^i, \quad (1)$$

where  $\mathbf{X}^i \in \mathbb{R}^{B \times T \times D}$  is the  $i$ -th layer output,  $\mathbf{y}_{\text{spk}_k} \in (0, 1)^{B \times T \times 1}$  is the corresponding speech activity for  $k$ -th speaker, and  $\mathbf{X}_{\text{inj}}^i$  is the output with the speaker information injected.  $B$ ,  $T$  and  $D$  stand for batch size, number of frames and feature dimension, respectively. Here,  $f_{\text{inj}}$  can be any module that injects the speaker information (*i.e.*, learnable speaker kernel) into the layer output. In this paper, two linear layers with

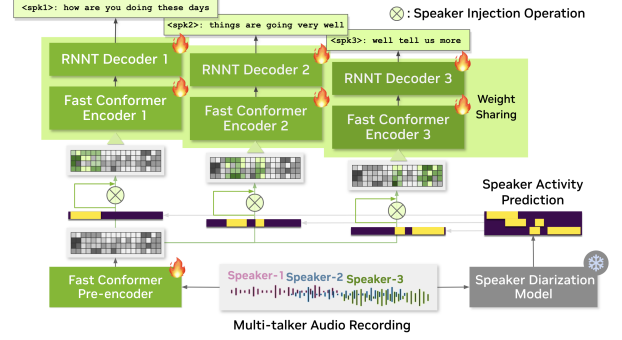


Figure 2: Multiple model instances are created and each model instance focuses on each speaker in the multi-talker recording.

an activation in between were applied for simplicity:

$$f_{\text{inj}}(\mathbf{X}^i, \mathbf{y}_{\text{spk}_k}) = f_{\text{feedforward}}(\mathbf{X}^i \odot \mathbf{y}_{\text{spk}_k}). \quad (2)$$

During the training stage, each input utterance contains overlapping speech from multiple speakers, and the speech activity of the target speaker is provided as an additional input. The target speaker is randomly selected from the set of speakers present in the utterance. Given the speech activity of the selected speaker, the model adapts to the target speaker autonomously, without requiring any pre-registered speaker profile. This design is particularly effective for streaming inference, where obtaining speaker profiles from overlapping speech is extremely difficult. The speech activities used during training can be obtained either from the ground truth labels or by applying a speaker diarization model to the input audio.

By leveraging the provided speech activities, the model dynamically adapts to the specific speaker, enabling robust recognition even in multi-speaker scenarios. Notably, the training process remains nearly identical to that of a conventional single-speaker ASR system, including the training objective and decoding procedure. The only distinctions lie in the model architecture, which includes the speaker injection module, and the use of multi-speaker speech data for training. This approach ensures compatibility with existing ASR frameworks while enhancing the model’s ability to handle target-speaker scenarios effectively.

### 2.2. Repurposing for Multi-Talker ASR

The proposed model is designed to adapt to a specific speaker when the corresponding speech activities are provided. This flexibility allows the model to serve dual purposes depending on the source of the speech activity information. If the speech activities are obtained from a personal VAD model [22], the model functions as a target-speaker ASR system, operating in a way that is consistent with the training process. Alternatively, when a speaker diarization system is used to generate speaker-specific speech activities, the model can be extended to a multi-speaker ASR system by running multiple instances in parallel, each focusing on a different speaker identified by the diarization output, as illustrated in Figure 2. Unlike conventional TS-ASR systems that require a speaker profile in advance, our model does not rely on any speaker identity information. Instead, it uses only the speech activity of the target speaker to adapt to that speaker during inference. This feature makes the model especially suitable for streaming applications, where acquiring a reliable speaker profile is often difficult or infeasible, particularly when the speaker begins with overlapped speech. As a result, the proposed method provides a practical and efficient

solution for both single-speaker and multi-speaker ASR in real-time scenarios.

Although the model is trained similarly to a single-speaker ASR system, meaning it can only focus on one speaker at a time during each inference step, it can be adapted to handle multiple speakers when multiple speech activities are available. These activities are typically provided by a speaker diarization model. To decode speech from multiple speakers simultaneously, multiple instances of the model can be employed. Specifically, a batch processing approach can be used, where the batch size corresponds to the number of speakers in the input audio. Each instance within the batch processes the speech activity of a distinct speaker, enabling the model to generate transcriptions for all speakers concurrently. This approach maintains the simplicity of the single-speaker ASR framework while extending its functionality to multi-talker scenarios.

Figure 3 shows the t-SNE plot of the ASR encoder state (activation at the last layer) for each token, where *2mix-spk0* and *2mix-spk1* are injected with the first and second speaker’s kernel, respectively. It is important to note that these ASR encoder representations, *2mix-spk0* and *2mix-spk1*, are derived from the same model and audio recording. We observe that the variability introduced by speaker injection is smaller than the distance between tokens; however, there is still a clear distinction between each speaker’s kernel, enabling the model to decode overlapping speech from two or three speakers.

### 2.3. Streaming Extension

The proposed method can also be extended to streaming scenarios, provided that both the ASR model and the speaker diarization model support streaming capabilities. In this work, we employ the FastConformer Transducer model [21] with cache-aware streaming [23] as the backbone for ASR training, ensuring efficient and low-latency processing of audio streams. In addition, we utilize the streaming Sortformer model [24], which is specifically designed for real-time end-to-end speaker diarization. By integrating these streaming-compatible models, the proposed method can be seamlessly adapted to streaming applications with minimal latency. This extension makes the system suitable for real-world applications such as live transcription, video conferencing, and other scenarios where low-latency processing is critical. The combination of streaming ASR and diarization models ensures that the system can handle continuous audio input while maintaining high accuracy.

## 3. Experiments and Results

### 3.1. Datasets and Evaluation Metrics

The training dataset was simulated using the LibriSpeech Corpus [25]. Since the model was trained with a single-speaker objective, alignment was not required for timestamp generation during training. For evaluation, we utilized the LibriSpeechMix dataset [12], which includes 1-mix, 2-mix, and 3-mix data to simulate single-speaker, two-speaker, and three-speaker scenarios, respectively. We also finetune the model on Fisher English Training Speech Part 1 and 2 [26] and evaluate it on CH109, a two-speaker subset of 109 sessions from the Callhome American English Speech (CHAES) dataset [27].

To evaluate the performance of the proposed model, we employed the concatenated minimum-permutation word error rate (cpWER) metric [28], which is commonly used for multi-talker ASR systems. This metric ensures a fair comparison by considering the best possible alignment between the predicted and

Table 1: DER results on LibriSpeechMix and CH109.

	Size	Latency (ms)	DER (%)		
			2-mix	3-mix	CH109
Sortformer	123M	$\infty$	2.5	3.2	-
		1120	3.3	4.1	5.6

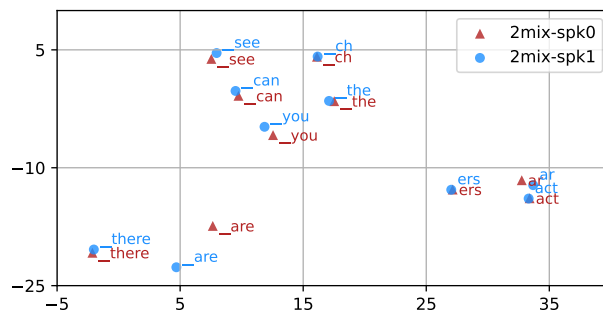


Figure 3: t-SNE plot of Fast Conformer’s last encoder states (embeddings) from an identical audio input (“there are the characters you can see”), using the same model weights but processed with different speaker kernels. The plot contrasts how ASR embeddings differ when generated using kernel *spk0* (triangles) versus *spk1* (circles), likely reflecting distinct speaker focuses.

reference transcriptions across all speakers.

### 3.2. Training Details

For the ASR model, we employed the FastConformer Transducer model [21, 23] and initialized it using the publicly available pre-trained model<sup>12</sup>. For the streaming diarization model, we use a fine-tuned version of the streaming Sortformer model<sup>3</sup> [29] on the simulated training set derived from the LibriSpeech Corpus.

The performance of the diarization model is reported in Table 1, which shows the Diarization Error Rate (DER) for LibriSpeechMix 2-mix, 3-mix with a collar of 0, and CH109 datasets with a collar of 0.25. For LibriSpeechMix evaluation, the silence at the beginning and end of each utterance is ignored.

During training, the diarization model was kept frozen and used to provide speech activities for each speaker. The speech activities were randomly sampled from one of the speakers in the input audio, and the corresponding transcriptions for that speaker were used as the training labels. This approach ensures that the model learns to adapt to the target speaker dynamically without requiring additional speaker-specific information. The top five checkpoints with the lowest validation word error rate (WER) are selected and averaged to produce the final checkpoint used for evaluation.

Both streaming and offline models were trained for 200k steps. For tokenization, we used SentencePiece [30] with byte pair encoding (BPE) and a vocabulary size of 1024. The tokenizer was trained on the training set of each dataset to ensure compatibility with the input data. The models were optimized using the AdamW optimizer [31] with a weight decay of 0.001 and the Noam learning rate scheduler [15] with a coefficient of

<sup>1</sup>Offline ASR model: [huggingface.co/nvidia/stt\\_en\\_fastconformer\\_transducer\\_large](https://huggingface.co/nvidia/stt_en_fastconformer_transducer_large)

<sup>2</sup>Streaming ASR model: [huggingface.co/nvidia/stt\\_en\\_fastconformer\\_hybrid\\_large\\_streaming\\_multi](https://huggingface.co/nvidia/stt_en_fastconformer_hybrid_large_streaming_multi)

<sup>3</sup>Streaming diarization model: [huggingface.co/nvidia/diar\\_streaming\\_sortformer\\_4spk-v2](https://huggingface.co/nvidia/diar_streaming_sortformer_4spk-v2)

Table 2: *Offline Max. 3-speaker systems on LibriSpeechMix.*

Systems	Model Size	cpWER (%)		
		1-mix	2-mix	3-mix
SOT [12]	135.6M	4.6	11.2	24.0
SOT-SQR [16]	136M	4.2	8.7	20.2
E2E-SA [19]	128.6M	3.3	4.3	6.0
Sidecar-Sep [32]	103.6M	-	5.7	-
MT-Whisper-L [33]	1.56B	-	3.4	6.8
DOM-SOT [17]	33M	5.2	5.6	10.0
SA-SOT [18]	136M	3.4	8.2	-
MT-LLM [34]	8B	2.3	5.2	10.2
AFT-MT [20]	156M	2.4	3.4	-
(Proposed) SSA	238M	<b>2.2</b>	<b>2.8</b>	<b>5.0</b>

5.0. All training runs were conducted with 8×NVIDIA Tesla A100 GPUs on one node.

### 3.3. Finetuning details

After training on the LibriSpeechMix dataset, we further fine-tune the model on the Fisher dataset for an additional 50k steps. Each session is truncated into segments ranging from 10 to 30 seconds, resulting in 64k training utterances and 1k validation utterances. The top five checkpoints with the lowest validation WER are selected and averaged to produce the final checkpoint used for evaluation.

### 3.4. Evaluation and Comparative Analysis

#### 3.4.1. Evaluation on LibriSpeechMix

Offline model is trained with 1-mix, 2-mix, and 3-mix simulated data, with a ratio of 1:3:6. We report the evaluations on 1-, 2-, and 3-mix data in Table 2. The proposed SSA model achieves state-of-the-art performance on the LibriSpeechMix dataset, with cpWERs of 2.2%, 2.8%, and 5.0% for 1-mix, 2-mix, and 3-mix scenarios, respectively. It outperforms most of the strong baselines with up to 3 speakers in the training data, particularly excelling in complex multi-talker settings. The model contains 238M parameters, which consist of a FastConformer Transducer ASR model (114M), a Sortformer diarization model (123M) and a speaker injection module (1.1M). The results highlight the effectiveness of the SSA mechanism in handling overlapping speech, making SSA a highly competitive and scalable solution for multi-talker ASR tasks.

For the streaming model, since there are no existing results for the 3-mix LibriSpeechMix dataset for comparison, we trained two distinct models. The first model is trained using 1-mix and 2-mix simulated data, with a ratio of 1:9 between 1-mix and 2-mix samples. The second model is trained using 1-mix, 2-mix, and 3-mix simulated data, with a ratio of 1:3:6. Table 3 presents the results of the model trained with up to 2 speakers, while Table 4 shows the results of the model trained with up to 3 speakers. The proposed SSA scheme demonstrates strong performance in streaming multi-talker ASR, with cpWERs of 4.0% (1-mix) and 5.6% (2-mix) at 560 ms latency, outperforming baselines under highly overlapped scenarios. Table 4 shows streaming multi-talker ASR with a maximum of 3 speakers with minor performance degradation from the 2-speaker model.

#### 3.4.2. Evaluation on Real data

We also evaluate the proposed model on real-life multi-speaker dataset CH109. The baseline is a cascade streaming multi-talker ASR model, which consists of a streaming single speaker

Table 3: *Streaming Max. 2-speaker systems on LibriSpeechMix.*

	Size	Latency(ms)		cpWER (%)	
		ASR	Diar.	1mix	2mix
Stream-T-SOT [35]	160M	160	2720	4.9	6.5
SSL-BLM-MT [36]	-	160	-	7.2	9.6
T-SOT-FNT [19]	-	160	-	4.7	10.1
AFT-MT [20]	156M	640	-	4.0	6.3
(Proposed) SSA	238M		80	6.8	8.3
			160	5.7	7.2
			560	<b>4.0</b>	<b>5.6</b>
			1120	3.8	5.2
			2720	3.4	4.6

Table 4: *Streaming Max. 3-speaker systems on LibriSpeechMix.*

	Size	Latency (ms)	cpWER (%)		
			1mix	2mix	3mix
(Proposed) SSA	238M	80	7.1	9.2	15.6
		160	6.2	7.8	15.7
		560	4.3	5.9	11.0
		1120	4.0	5.4	9.8
		2720	3.7	4.9	8.1

Table 5: *Streaming Max. 2-speaker systems on CH109.*

	Size	Latency (ms)	cpWER (%)
LLM-BSD [37]	2.2B	∞	26.31
Cascaded Model	238M	1120	27.38
(Proposed) SSA	238M	1120	26.21

ASR model and a streaming diarization model. In the cascaded model, words and speaker segments are matched using time-stamps. The ASR model for the cascaded baseline is the same as the model whose parameters are employed for initialization for streaming model training, as mentioned in Section 3.2, and the streaming diarization model for the baseline is the same as the model used for streaming SSA inference, as mentioned in Section 3.3. The proposed SSA model demonstrates superior performance on the CH109 dataset, achieving a cpWER of 26.21% at a latency of 1120 ms. This outperforms not only the cascaded streaming model but also the performance of the offline system [37], despite our proposed system operating under low-latency streaming constraints.

## 4. Conclusion

In this paper, we introduced a query-less speaker targeting approach that employs a multi-instance encoder-decoder for each speaker. Following the design principle of maximizing the performance of the base monaural ASR system, the proposed multi-instance speaker targeting approach shows that the relatively long frame length can be addressed by employing multiple instances of single-speaker ASR models. Our method achieves state-of-the-art performance in both streaming and offline setups on the LibriSpeechMix dataset, effectively addressing the challenges of multi-speaker scenarios and overlapping speech. Future work includes the application of the proposed method on Transformer-based ASR systems and multi-modal large language models (LLMs), thus equipping the foundational ASR models or LLMs with the state-of-the-art multi-talker ASR capability.

## 5. References

- [1] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013, pp. 55–59.
- [2] 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, 2010.
- [3] O. Abdel-Hamid and H. Jiang, "Fast speaker adaptation of hybrid nn/hmm model for speech recognition based on discriminative learning of speaker code," in *Proc. ICASSP*, 2013, pp. 7942–7946.
- [4] L. Lee and R. C. Rose, "Speaker normalization using efficient frequency warping procedures," in *Proc. ICASSP*, vol. 1, 1996, pp. 353–356.
- [5] P. Swietojanski and S. Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *2014 IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 171–176.
- [6] C. Zhang and P. C. Woodland, "Parameterised sigmoid and relu hidden activation functions for dnn acoustic modelling," in *Sixteenth annual conference of the international speech communication association*, 2015.
- [7] Y. Zhao, J. Li, and Y. Gong, "Low-rank plus diagonal adaptation for deep neural networks," in *Proc. ICASSP*, 2016, pp. 5005–5009.
- [8] M. K. Baskar, T. Herzig, D. Nguyen, M. Diez, T. Polzehl, L. Burget, J. Černocký *et al.*, "Speaker adaptation for wav2vec2 based dysarthric asr," *arXiv preprint arXiv:2204.00770*, 2022.
- [9] J. Deng, X. Xie, T. Wang, M. Cui, B. Xue, Z. Jin, M. Geng, G. Li, X. Liu, and H. Meng, "Confidence score based conformer speaker adaptation for speech recognition," in *Proc. INTERSPEECH*, 2022, pp. 2623–2627.
- [10] I. Medennikov, M. Korenevsky, T. Prisyach, Y. Khokhlov, M. Korenevskaya, I. Sorokin, T. Timofeeva, A. Mitrofanov, A. Andrusenko, I. Podluzhny *et al.*, "The STC System for the CHiME-6 Challenge," in *CHiME 2020 Workshop on Speech Processing in Everyday Environments*, 2020.
- [11] C. Boeddeker, J. Heitkaemper, J. Schmalenstroer *et al.*, "Front-End Processing for the CHiME-5 Dinner Party Scenario," in *CHiME5 Workshop*, 2018.
- [12] N. Kanda, Y. Gaur *et al.*, "Serialized Output Training for End-to-End Overlapped Speech Recognition," in *Proc. INTERSPEECH*, 2020, pp. 2797–2801.
- [13] X. Chang, Y. Qian *et al.*, "End-to-End Monaural Multi-Speaker ASR System without Pretraining," in *Proc. ICASSP*, 2019, pp. 6256–6260.
- [14] X. Chang, W. Zhang *et al.*, "MIMO-SPEECH: End-to-end Multi-Channel Multi-Speaker Speech Recognition," in *Proc. ASRU*, 2019, pp. 237–244.
- [15] A. Vaswani, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.
- [16] N. Kanda, Y. Gaur, X. Wang *et al.*, "Joint Speaker Counting, Speech Recognition, and Speaker Identification for Overlapped Speech of Any Number of Speakers," in *Proc. INTERSPEECH*, 2020.
- [17] Y. Shi, L. Li, S. Yin, D. Wang, and J. Han, "Serialized output training by learned dominance," *arXiv preprint arXiv:2407.03966*, 2024.
- [18] Z. Fan, L. Dong, J. Zhang, L. Lu, and Z. Ma, "Sa-sot: Speaker-aware serialized output training for multi-talker asr," in *Proc. ICASSP*, 2024, pp. 9986–9990.
- [19] N. Kanda, G. Ye, Y. Gaur, X. Wang, Z. Meng, Z. Chen, and T. Yoshioka, "End-to-end speaker-attributed asr with transformer," *arXiv preprint arXiv:2104.02128*, 2021.
- [20] T. Moriya, S. Horiguchi, M. Delcroix, R. Masumura, T. Ashihara, H. Sato, K. Matsuura, and M. Mimura, "Alignment-free training for transducer-based multi-talker asr," *arXiv preprint arXiv:2409.20301*, 2024.
- [21] D. Rekish, N. R. Koluguri, S. Kriman, S. Majumdar, V. Noroozi, H. Huang, O. Hrinchuk, K. Puvvada, A. Kumar, J. Balam *et al.*, "Fast conformer with linearly scalable attention for efficient speech recognition," in *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2023, pp. 1–8.
- [22] S. Ding, Q. Wang, S.-Y. Chang, L. Wan, and I. Lopez Moreno, "Personal vad: Speaker-conditioned voice activity detection," in *The Speaker and Language Recognition Workshop (Odyssey 2020)*, 2020, pp. 433–439.
- [23] V. Noroozi, S. Majumdar, A. Kumar, J. Balam, and B. Ginsburg, "Stateful conformer with cache-based inference for streaming automatic speech recognition," in *Proc. ICASSP*, 2024, pp. 12 041–12 045.
- [24] I. Medennikov, T. Park, W. Wang, H. Huang, K. Dhawan, J. Wang, J. Balam, and B. Ginsburg, "Streaming Sortformer: Speaker Cache-Based Online Speaker Diarization with Arrival-Time Ordering," in *Proc. INTERSPEECH*, 2025.
- [25] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Proc. ICASSP*, 2015, pp. 5206–5210.
- [26] C. Cieri, D. Miller, and K. Walker, "The Fisher Corpus: A Resource for the Next Generations of Speech-to-text," in *Proc. LREC*, 2004, pp. 69–71.
- [27] A. Canavan, D. Graff, and G. Zipperlen, "Callhome american english speech," Web Download, Philadelphia, 1997, IDC97S42. [Online]. Available: <https://catalog.ldc.upenn.edu/LDC97S42>
- [28] S. Watanabe, M. Mandel, J. Barker *et al.*, "CHiME-6 Challenge: Tackling Multispeaker Speech Recognition for Unsegmented Recordings," in *CHiME Workshop*, 2020.
- [29] T. Park, I. Medennikov, K. Dhawan, W. Wang, H. Huang, N. R. Koluguri, K. C. Puvvada, J. Balam, and B. Ginsburg, "Sortformer: Seamless integration of speaker diarization and asr by bridging timestamps and tokens," *arXiv preprint arXiv:2409.06656*, 2024.
- [30] T. Kudo and J. Richardson, "SentencePiece: A Simple and Language Independent Subword Tokenizer and Detokenizer for Neural Text Processing," in *EMNLP: System Demonstrations*, 2018.
- [31] I. Loshchilov, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.
- [32] L. Meng, J. Kang, M. Cui, Y. Wang, X. Wu, and H. Meng, "A sidcar separator can convert a single-talker speech recognition system to a multi-talker one," in *Proc. ICASSP*, 2023, pp. 1–5.
- [33] L. Meng, J. Kang, Y. Wang, Z. Jin, X. Wu, X. Liu, and H. Meng, "Empowering whisper as a joint multi-talker and target-talker speech recognition system," in *Proc. INTERSPEECH*, 2024, pp. 4653–4657.
- [34] L. Meng, S. Hu, J. Kang, Z. Li, Y. Wang, W. Wu, X. Wu, X. Liu, and H. Meng, "Large language model can transcribe speech in multi-talker scenarios with versatile instructions," *arXiv preprint arXiv:2409.08596*, 2024.
- [35] N. Kanda, J. Wu, Y. Wu, X. Xiao, Z. Meng, X. Wang, Y. Gaur, Z. Chen, J. Li, and T. Yoshioka, "Streaming speaker-attributed asr with token-level speaker embeddings," in *Proc. INTERSPEECH*, 2022, pp. 521–525.
- [36] Z. Huang, Z. Chen, N. Kanda, J. Wu, Y. Wang, J. Li, T. Yoshioka, X. Wang, and P. Wang, "Self-supervised learning with bi-label masked speech prediction for streaming multi-talker speech recognition," in *Proc. ICASSP*, 2023, pp. 1–5.
- [37] T. J. Park, K. Dhawan, N. Koluguri, and J. Balam, "Enhancing Speaker Diarization with Large Language Models: A Contextual Beam Search Approach," in *Proc. ICASSP*, 2024, pp. 10 861–10 865.