

CMT-LLM: Contextual Multi-Talker ASR Utilizing Large Language Models

Jiajun He^{1,2}, Naoki Sawada², Koichi Miyazaki², Tomoki Toda³

¹Graduate School of Informatics, Nagoya University, Japan

²AI Lab, CyberAgent, Japan

³Information Technology Center, Nagoya University, Japan

jiajun.he@g.sp.m.is.nagoya-u.ac.jp, sawada_naoki@cyberagent.co.jp,
miyazaki_koichi_xa@cyberagent.co.jp, tomoki@icts.nagoya-u.ac.jp

Abstract

In real-world applications, automatic speech recognition (ASR) systems must handle overlapping speech from multiple speakers and recognize rare words like technical terms. Traditional methods address multi-talker ASR and contextual biasing separately, limiting performance in complex scenarios. We propose a unified framework that combines multi-talker overlapping speech recognition and contextual biasing into a single task. Our ASR method integrates pretrained speech encoders and large language models (LLMs), using optimized finetuning strategies. We also introduce a two-stage filtering algorithm to efficiently identify relevant rare words from large biasing lists and incorporate them into the LLM's prompt input, enhancing rare word recognition. Experiments show that our approach outperforms traditional contextual biasing methods, achieving a WER of 7.9% on LibriMix and 32.9% on AMI SDM when the biasing size is 1,000, demonstrating its effectiveness in complex speech scenarios.

Index Terms: Multi-talker ASR, Contextual ASR, Large Language Models, Serialized Output Training, Contextual Biasing

1. Introduction

Multi-talker automatic speech recognition (ASR), particularly in overlapping speech scenarios, remains a major challenge. Existing methods include permutation invariant training [1], heuristic error assignment [2], and serialized output training (SOT) [3]. Among these, SOT has gained attention for resolving speaker arrangement uncertainty by concatenating transcriptions in speech order. However, it relies heavily on long-context modeling, where attention-based encoder-decoder (AED) models struggle with inter-speaker dependencies.

Another key challenge in ASR is recognizing rare words, such as proper names and technical terms, which are underrepresented in training data [4]. Contextual biasing methods address this by incorporating external biasing lists, but existing approaches have limitations. Shallow fusion adjusts decoding scores but struggles with large lists and dynamic contexts [5]. Deep biasing [6] improves accuracy by integrating biasing features but requires retraining and architectural modifications. Deep context models [7] leverage contextual text encoders but are computationally intensive. Contextual ASR error correction [8, 9, 10] refines outputs post-recognition but depends on initial ASR hypotheses, introducing latency.

In real-world applications, multi-talker ASR and contextual biasing often arise as intertwined challenges. For example, in meeting transcription, the ASR system needs to both distinguish between speakers and accurately recognize domain-

specific terms and proper nouns. Similarly, in customer service settings with concurrent conversations, the system must dynamically identify each speaker's contributions while incorporating contextual information to enhance decoding accuracy. Despite their inherent interconnection, existing research typically treats multi-talker ASR and contextual biasing as separate tasks. To bridge this gap, we propose a novel task: the integrated application of contextual biasing within multi-talker ASR, aimed at improving the recognition of rare words in overlapping speech scenarios. To the best of our knowledge, this is the first study to combine multi-talker ASR with contextual biasing.

In recent years, the application of large language models (LLMs) in speech processing has garnered increasing attention [11, 12, 13, 14, 15]. With their remarkable global modeling capabilities, LLMs can effectively capture inter-speaker dependencies in multi-talker scenarios, producing semantically coherent and natural transcription outputs [13, 14]. In the context of contextual biasing, LLMs leverage prompt-based learning to flexibly integrate dynamic biasing lists provided by users, enabling accurate recognition of rare words while significantly reducing reliance on complex decoding mechanisms [15]. Consequently, LLMs demonstrate unique advantages in addressing the dual challenges of multi-talker ASR and contextual biasing.

To achieve this, we propose a comprehensive framework integrating a pretrained speech encoder, projector, and LLM. Specifically, we adopt a two-stage finetuning strategy to address the challenges posed by the multi-talker ASR task and the incorporation of large biasing lists in a practical setting: first, the self-supervised learning (SSL) speech encoder is finetuned using the conventional SOT method. Next, we freeze the finetuned SSL speech encoder and LLM, training only the projector while efficiently finetuning the LLM using low-rank adaptation (LoRA). Moreover, considering the challenges posed by large biasing lists (e.g., thousands of words) in practical applications—where LLM's prompt-based learning cannot effectively handle such extensive vocabularies—we filter and select the most relevant rare words based on coarse decoding results from the first stage. These refined small-scale biasing lists are then incorporated into the LLM's prompt input, effectively addressing the challenges of large biasing lists and significantly improving contextual biasing performance, ultimately improving ASR accuracy.

2. Proposed Method

2.1. Problem Formulation

The contextual multi-talker ASR problem can be formalized as the mapping function $f(S, C) = T$, where the speech $S = (S_1, S_2, \dots, S_M)$ contains M acoustic frames, the context $C = (C_1, C_2, \dots, C_L) \in \mathbb{R}^L$ denotes the biasing list containing L rare words, and the ground truth transcript $T =$

This work was done during Jiajun He's internship at CyberAgent.

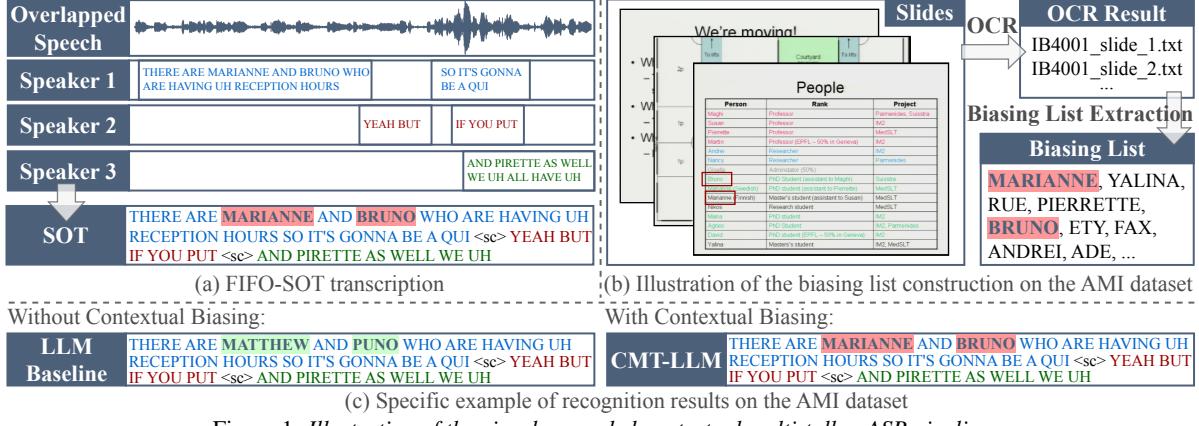


Figure 1: Illustration of the visual-grounded contextual multi-talker ASR pipeline.

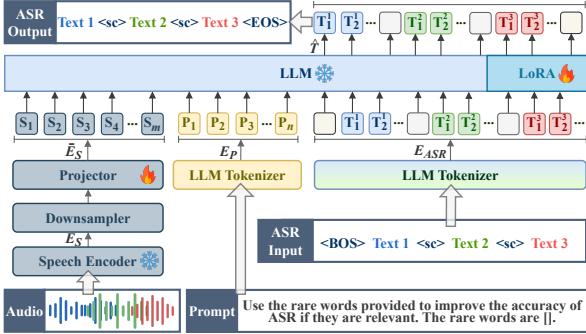


Figure 2: Overall architecture of the CMT-LLM model.

$(T_1^1, T_2^1, \dots, \text{<sc>} T_1^2, T_2^2, \dots, \text{<sc>} T_1^3, T_2^3, \dots) \in \mathbb{R}^N$ is the token sequence. In T , T_a^b represents the a -th token of the b -th speaker and N denotes the length of the transcript. Following previous works [16, 3], we use the SOT method to address the multi-talker ASR problem. Specifically, assuming the number of speakers is 3, the transcriptions of different speakers are concatenated by inserting the speaker change symbol “ <sc> ”, creating the reference transcription for overlapping speech. The concatenation order follows the speaking time of each speaker, known as first-in first-out (FIFO), as shown in Fig. 1(a).

2.2. Proposed CMT-LLM

In this section, we introduce a contextual multi-talker ASR method with LLM (**CMT-LLM**). As illustrated in Fig. 2, like most previous LLM studies [12, 13, 14, 15, 17], the proposed architecture comprises three key components: a speech encoder, a linear projector, and an LLM decoder. First, the speech encoder processes the overlapping speech signal and extracts the corresponding speech representation $E_S \in \mathbb{R}^{M \times d_S}$, where d_S represents the dimension of the extracted speech features. To make the representation more manageable for the LLM, we apply a 1D convolutional layer for downsampling, reducing the temporal resolution by a factor of n . The resulting features are then passed through a projection module consisting of two linear layers, transforming them into a speech embedding $\bar{E}_S \in \mathbb{R}^{\frac{M}{n} \times d_P}$. Here, d_P denotes the dimension of the projected speech embedding, which matches the hidden size of the LLM to ensure compatibility with its input format.

For contextual multi-talker ASR, we enhance the model’s performance by incorporating a biasing list of rare words into the LLM’s prompt, enabling more accurate recognition of infrequent and potentially error-prone vocabulary. This prompt, which includes task-specific instructions and supplementary

contextual information, undergoes tokenization and encoding to produce the prompt text embedding $E_P \in \mathbb{R}^{P \times d_P}$, where P represents the length of the prompt. The process for generating the biasing list is detailed in Section 3.2.

During training, the SOT-style multi-talker transcript is tokenized to obtain the ASR embedding $E_{ASR} \in \mathbb{R}^{N \times d_P}$. This embedding, together with the speech embedding \bar{E}_S and the prompt text embedding E_P , is provided as input to the LLM, which is trained to predict the target transcript $\hat{T} \in \mathbb{R}^N$. The model’s performance is optimized by minimizing the cross-entropy loss between the predicted transcript \hat{T} and the ground truth T . During inference, both the speech embedding and the prompt text embedding (including task-specific instructions and the biasing list) are provided to the LLM, which then generates the ASR transcript in an autoregressive manner. To evaluate the impact of contextual information, we also assess a variant of the model without the biasing list in the prompt, referred to as the LLM Baseline.

The biasing list size in the LLM prompt is limited to 100 words to minimize computational costs during training. The detailed construction process will be described in Section 3.2. However, in real-world applications, the biasing list may contain tens of thousands of words in the inference stage, which challenges the CMT-LLM’s ability to select the most relevant words effectively, causing significant performance degradation.

To address this issue, we introduce a two-stage filtering approach that targets the challenges of large biasing lists inspired by previous studies [18, 19, 20, 21] during inference. Specifically, in the first stage, we finetune a SSL pretrained speech model on the target dataset, adding a simple CTC head. A greedy decoding algorithm is then employed to generate initial predictions with lower computational overhead. The aims of this step are: 1) Previous research indicates that finetuning a pretrained speech encoder using traditional methods before applying LLM-based ASR training typically outperforms using the pretrained speech encoder directly; 2) The initial decoding results help filter out irrelevant rare words from the large biasing list.

In the second stage, we process the initial CTC decoding results by first removing the most common 5,000 words to retain more distinctive rare words. For example, if the ground truth transcription is “... MORE THAN THE SPEAKER CHARACTERISATION AS M STEVE ...” and the initial decoding is “... MORE THAN THE SPEAKER CHARACE THSATION AS STEE ...”, after removing common words, the remaining words are “CHARACE THSATION” and “STEE”. Next, we

generate all possible subcombinations of these remaining words to more thoroughly select the most relevant terms. For example, given “CHARACE THSATION”, the possible segments include “CHARACE”, “THSATION”, and “CHARACE THSATION”. We calculate the word-based edit distance between these segments and the words in the biasing list, selecting the closest match, “CHARACTERISATION”. We notice that phoneme-based edit distance calculation and text semantic similarity calculation are both slower and less effective than the simplest word-based edit distance calculation. If only the individual words “CHARACE” and “THSATION” are considered, the target word “CHARACTERISATION” might be overlooked. To balance accuracy, efficiency, and computational cost, we match each candidate word with its Top-10 related words during filtering. Finally, we merge the selected relevant rare words, remove duplicates, and add them to the LLM prompt.

3. Experimental Evaluation

3.1. Implementation Details

Our method was trained on 4 NVIDIA A100 80 GB GPUs and the batch size was set to 2. We employed the finetuned WavLM-Large [22] as the speech encoder, processing 16 kHz sampled audio into feature embeddings with a frame rate of 50 Hz and a dimension of 1,024. These embeddings underwent downsampling ($n = 5$) and were transformed via two linear projection layers. This process produced speech embeddings with a final frame rate of 10 Hz and a dimension of 4,096.

For the LLM module, we integrated Vicuna-7B [23], a variant of LLaMA [24] finetuned on ShareGPT conversational data. During training, only the projection layer was trained, while the speech encoder and LLM remained frozen. We adopted the AdamW optimizer [25] with a learning rate of 0.0001, hyperparameters β set to (0.9, 0.999), epsilon to 1e-08, and weight decay to 1e-6. The training strategy followed a linear warmup schedule with 1,000 warmup steps, continuing for up to 100,000 steps, with early stopping triggered by stagnating validation loss. Additionally, LoRA was applied for LLM finetuning, with α set to 32, r set to 8, and dropout set to 0.05. We defined specific prompt formats for different models. The prompt for the LLM Baseline was simply: “Transcribe speech to text.”. In contrast, the CMT-LLM employed a more detailed prompt with contextual information: “Use the rare words provided to improve the accuracy of ASR if they are relevant. The rare words are [...].”, where the biasing word list is dynamically inserted into the brackets. During inference, we utilized beam search decoding with a beam size of 4.

Table 1: WER performance comparison with different multi-talker ASR models on LibriMix (%) with 1,000 distractors.

Model	Year	LibriMix	
		Dev	Test
Conditional-Conformer-CTC [26]	2021	24.5	24.9
WavLM-CTC [22]	2022	23.0	20.3
Whisper (Small)	2023	26.0	25.0
Conformer + WavLM-Large upstream	2022	24.7	23.3
GEncSep [3]	2023	19.4	17.1
LLM Baseline	2024	17.2	15.0
CMT-LLM (ours)	2025	8.1	7.3

<https://huggingface.co/microsoft/wavlm-large>
<https://huggingface.co/lmsys/vicuna-7b-v1.5>

Table 2: WER performance comparison with different multi-talker ASR models on AMI (%) with 1,000 distractors.

Model	Year	AMI					
		IHM-Mix		SDM		MDM	
		Dev	Test	Dev	Test	Dev	Test
WavLM-CTC [22]	2022	34.4	34.3	39.8	44.0	38.1	41.5
SURT 2.0 (Large) [27] + Adaptation [27]	2023	-	36.8	-	62.5	-	44.4
	2023	-	35.1	-	44.6	-	41.4
LLM Baseline	2025	24.1	23.5	30.9	34.2	32.9	31.2
CMT-LLM (ours)	2025	21.9	22.8	30.0	32.9	29.7	30.4

3.2. Experimental Conditions

Datasets: We evaluate the model using the LibriMix [28] and AMI [29] datasets:

- The **LibriMix** dataset [28] combines LibriSpeech [30] speech with WHAM! noise [31]. Following the ESPNet SOT, we introduce a random delay (1.0–1.5s) between overlapping speakers, creating two-speaker mixtures. The dataset includes 830h of speed-perturbed training data, 8.2h validation, and 7.6h test data.
- The **AMI** dataset [29] contains 100h of meeting recordings with 4–5 speakers. Following the icefall SURT, we use three microphone settings: IHM-Mix (mixed headset mics), SDM (single distant mic), and MDM (beamformed array) [32]. It includes 79.4h training, 9.7h validation, and 9.1h test data.

Biasing List Construction: Since Librimix lacks predefined biasing lists, we follow the validated simulation method in [33]. As Librimix is derived from Librispeech, we use the same full biasing list, containing 209.2K words. Words in this list are considered rare. For each utterance, we construct biasing lists by selecting words from reference transcripts that appear in the full list, adding distractors as per the experimental setup. To assess real-world feasibility, we construct AMI biasing lists by extracting text from lecture slides using Tesseract OCR, as shown in Fig. 1 (b). Unique words are extracted, and those in the full biasing list or occurring fewer than 100 times are classified as rare. These lecture-specific lists are used for contextual correction [34]. Additionally, we merge all lecture lists into a unified AMI biasing list for large-scale experiments, selecting distractors accordingly. Note that the total number of words in the biasing lists of AMI accounts for about 1.0% of the total vocabulary, which has a small impact on the overall WER. However, as shown in the example in Fig. 1, these words are mostly crucial content words, and their correct recognition is critical for understanding the utterance.

Evaluation Metrics: We evaluate the ASR performance using word error rate (WER) and biased WER (B-WER) [33], which computes the WER based on words present in the biasing list.

3.3. Results and Analysis

Comparisons of Baselines: Table 1 and Table 2 compare WER performance across different multi-talker ASR models on LibriMix and AMI datasets. CMT-LLM achieves the lowest WER in both cases, significantly outperforming the LLM Baseline and other SOTA traditional models, demonstrating its effectiveness in contextual multi-talker ASR.

Table 3 compares different contextual ASR models. The LLM baseline, lacking a biasing list, struggles with high B-WER, highlighting the challenge of rare word recognition with-

https://github.com/espnet/espnet/tree/master/egs2/librimix/sot_asr1
<https://github.com/k2-fsa/icefall/tree/master/egs/ami/SURT>
https://github.com/facebookresearch/fb.ai-speech/tree/master/is21_deep_bias
<https://github.com/tesseract-ocr/tesseract>

Table 3: WER performance comparison with different contextual ASR models on LibriMix and AMI test sets (%).

Model	Type	Distractors	LibriMix		AMI		
			WER / B-WER	WER / B-WER	IHM-Mix	SDM	MDM
LLM Baseline	No Biasing	-	9.2 / 25.3	23.5 / 45.6	34.2 / 51.0	31.2 / 49.7	
+ ED-CEC [8]	Biasing List	+ 100	8.4 / 15.3	23.1 / 34.2	33.6 / 35.5	30.8 / 32.1	
		+ 1,000	8.7 / 15.5	23.1 / 34.8	33.7 / 36.9	30.8 / 35.8	
		+ 2,000	8.8 / 16.0	23.2 / 35.1	33.7 / 37.7	30.9 / 38.7	
		+ 5,000	8.9 / 16.4	23.2 / 35.5	33.8 / 39.0	31.0 / 40.3	
CMT-LLM	No Biasing	-	9.3 / 28.7	23.4 / 48.7	33.6 / 50.8	31.1 / 49.5	
	Anti-Context	+ 100	9.4 / 29.7	23.3 / 43.8	33.6 / 50.2	30.9 / 48.2	
	Biasing List	+ 100	7.3 / 8.7	22.7 / 29.8	32.7 / 30.6	30.3 / 30.2	
		+ 1,000	7.9 / 12.5	22.8 / 33.6	32.9 / 35.0	30.4 / 34.6	
		+ 2,000	8.4 / 14.1	22.8 / 35.1	33.0 / 38.9	30.5 / 36.7	
		+ 5,000	8.4 / 15.2	22.9 / 36.4	33.1 / 42.5	30.6 / 38.1	
	GT Rare Words	-	6.6 / 2.6	22.4 / 24.6	31.7 / 28.4	29.5 / 28.6	

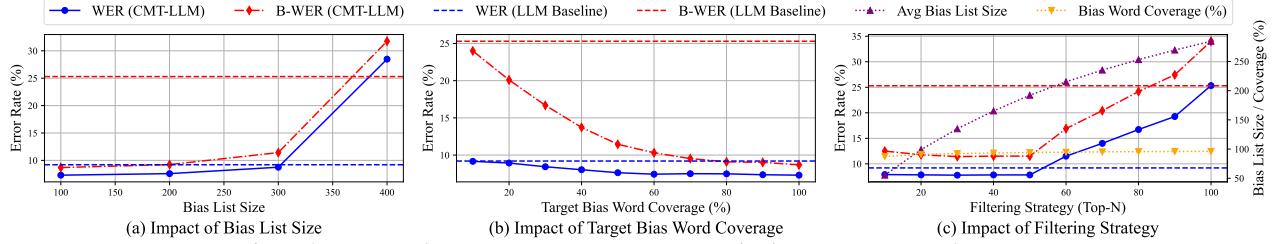


Figure 3: Impact of Biasing List Size, Coverage, and Filtering on ASR Performance.

out explicit context. ED-CEC [8], a SOTA contextual ASR post-processing method, improves B-WER with a small biasing list (+100 distractors) but becomes less effective as the list size increases, owing to growing ambiguity from additional distractors. In contrast, CMT-LLM, which integrates the biasing list directly into the prompt, performs significantly better with small biasing lists, surpassing both the LLM baseline and ED-CEC by a large margin. In the Anti-Context condition, where target bias words are substituted with distractors, B-WER increases significantly, highlighting the crucial role of explicit biasing. The GT Rare Words condition, which biases only the ground-truth rare words, serves as an ideal upper bound, achieving a B-WER of just 2.6% on LibriMix, showcasing the effectiveness of optimized biasing strategies. However, recognition performance declines when the biasing list exceeds 1,000 words, not only owing to more distractors but also reduced target biasing word coverage from filtering. Specifically, after adding 1,000, 2,000, and 5,000 distractors, coverage drops to 87.40%, 85.07%, and 83.07%, respectively. Although filtering helps CMT-LLM handle large biasing lists, fewer target words negatively impact recognition. Overall, at the same target biasing word coverage, incorporating biasing information in the prompt is more effective than post-processing correction.

Impact of Biasing List Size and Word Coverage: Finally, we examine how biasing list size and word coverage impact recognition performance (Fig. 3). Without a filtering mechanism (Fig. 3(a)), increasing the biasing list size from 100 to 400 significantly raises WER and B-WER. When the list exceeds 300, WER surpasses the LLM Baseline, suggesting that an overly large list introduces interference and degrades performance. With the list size fixed at 100 (Fig. 3(b)), reducing word coverage from 100% to 10% sharply increases B-WER,

indicating that low coverage weakens the biasing effect. By applying a filtering mechanism to an initial list containing 1,000 distractors (Fig. 3(c)), we find that if the average size of the filtered biasing list is below 200, high coverage can be maintained while minimizing interference. However, when Top-N exceeds 50, WER and B-WER spike, and at Top-60, WER surpasses the LLM Baseline, confirming that excessively large lists impair recognition. Future work focuses on reducing list size while maintaining high coverage to improve system robustness. **Example of Recognizing Rare Words:** Fig. 1(c) provides a specific example of CMT-LLM using rare personal names extracted from slides to improve ASR accuracy, demonstrating the effectiveness of the proposed CMT-LLM.

4. Conclusion

This paper for the first time introduces an LLM-based SOT method CMT-LLM for multi-talker contextual ASR. Leveraging its strong decoding capabilities, deep comprehension of long-range context, and cross-speaker modeling ability, LLM excels in handling complex multi-talker speech scenarios. In addition, CMT-LLM effectively incorporates contextual information through prompt learning, leading to a significant improvement in recognizing rare words. Specifically, for large biasing lists containing thousands of words, a coarse decoding-based filtering algorithm is applied to substantially reduce the list size, preserving only the 10 most relevant words for each remaining word and integrating them into the prompt input. This further enhances the contextual biasing effect even with 5,000 distractors. Experimental results demonstrate that our approach achieves state-of-the-art recognition performance on both the simulated LibriMix dataset and the real-world AMI dataset, significantly outperforming existing conventional methods.

5. Acknowledgments

This work was partly supported by JST CREST Grant Number JPMJCR22D1, Japan.

6. References

- [1] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, “Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks,” *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 25, no. 10, pp. 1901–1913, 2017.
- [2] L. Lu, N. Kanda, J. Li, and Y. Gong, “Streaming end-to-end multi-talker speech recognition,” *IEEE Signal Processing Letters*, vol. 28, pp. 803–807, 2021.
- [3] H. Shi, Y. Gao, Z. Ni, and T. Kawahara, “Serialized speech information guidance with overlapped encoding separation for multi-speaker automatic speech recognition,” in *Proc. SLT*, 2024, pp. 1–7.
- [4] J. He, Z. Yang, and T. Toda, “Enhancing recognition of rare words in ASR through error detection and context-aware error correction,” *IEICE Tech. Rep.*, vol. 123, no. 292, pp. 13–18, 2023.
- [5] D. Le, G. Keren, J. Chan, J. Mahadeokar, C. Fuegen, and M. L. Seltzer, “Deep shallow fusion for RNN-T personalization,” in *Proc. SLT*, 2021, pp. 251–257.
- [6] G. Sun, C. Zhang, and P. C. Woodland, “Graph neural networks for contextual ASR with the tree-constrained pointer generator,” *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 32, pp. 2407–2417, 2024.
- [7] K. Huang, A. Zhang, Z. Yang, P. Guo, B. Mu, T. Xu, and L. Xie, “Contextualized end-to-end speech recognition with contextual phrase prediction network,” in *Proc. Interspeech*, 2023, pp. 4933–4937.
- [8] J. He, Z. Yang, and T. Toda, “ED-CEC: Improving rare word recognition using ASR postprocessing based on error detection and context-aware error correction,” in *Proc. ASRU*, 2023, pp. 1–6.
- [9] J. He, X. Shi, X. Li, and T. Toda, “MF-AED-AEC: Speech emotion recognition by leveraging multimodal fusion, ASR error detection, and ASR error correction,” in *Proc. ICASSP*, 2024, pp. 11 066–11 070.
- [10] J. He and T. Toda, “PMF-CEC: Phoneme-augmented multimodal fusion for context-aware ASR error correction with error-specific selective decoding,” *arXiv preprint arXiv:2025*, 2025.
- [11] Y. Bai, J. Chen, J. Chen, W. Chen, Z. Chen, C. Ding, L. Dong, Q. Dong, Y. Du, K. Gao *et al.*, “Seed-ASR: Understanding diverse speech and contexts with LLM-based speech recognition,” in *arXiv preprint arXiv:2407.04675*, 2024.
- [12] Z. Ma, G. Yang, Y. Yang, Z. Gao, J. Wang, Z. Du, F. Yu, Q. Chen, S. Zheng, S. Zhang *et al.*, “An embarrassingly simple approach for LLM with strong ASR capacity,” in *arXiv preprint arXiv:2402.08846*, 2024.
- [13] 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,” in *Proc. ICASSP*, 2025, pp. 1–5.
- [14] M. Shi, Z. Jin, Y. Xu, Y. Xu, S.-X. Zhang, K. Wei, Y. Shao, C. Zhang, and D. Yu, “Advancing multi-talker ASR performance with large language models,” in *Proc. SLT*, 2024, pp. 14–21.
- [15] G. Yang, Z. Ma, F. Yu, Z. Gao, S. Zhang, and X. Chen, “Malaa-ASR: Multimedia-assisted LLM-based ASR,” *Proc. Interspeech*, pp. 2405–2409, 2024.
- [16] N. Kanda, Y. Gaur, X. Wang, Z. Meng, and T. Yoshioka, “Serialized output training for end-to-end overlapped speech recognition,” in *Proc. Interspeech*, 2020, pp. 2797–2801.
- [17] S. Hu, L. Zhou, S. Liu, S. Chen, L. Meng, H. Hao, J. Pan, X. Liu, J. Li, S. Sivasankaran, L. Liu, and F. Wei, “WavLLM: Towards robust and adaptive speech large language model,” in *Proc. EMNLP*, 2024, pp. 4552–4572.
- [18] Z. Yang, S. Sun, X. Wang, Y. Zhang, L. Ma, and L. Xie, “Two stage contextual word filtering for context bias in unified streaming and non-streaming transducer,” in *Proc. Interspeech*, 2023, pp. 3257–3261.
- [19] K. Huang, A. Zhang, Z. Yang, P. Guo, B. Mu, T. Xu, and L. Xie, “Contextualized end-to-end speech recognition with contextual phrase prediction network,” in *Proc. Interspeech*, 2023, pp. 4933–4937.
- [20] T. Xu, Z. Yang, K. Huang, P. Guo, A. Zhang, B. Li, C. Chen, C. Li, and L. Xie, “Adaptive contextual biasing for transducer based streaming speech recognition,” in *Proc. Interspeech*, 2023, pp. 1668–1672.
- [21] G. Yang, Z. Ma, Z. Gao, S. Zhang, and X. Chen, “CTC-assisted LLM-based contextual ASR,” in *Proc. SLT*, 2024, pp. 126–131.
- [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] W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J. E. Gonzalez *et al.*, “Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality,” See <https://vicuna.lmsys.org> (accessed 14 April 2023), vol. 2, no. 3, p. 6, 2023.
- [24] H. Touvron, T. Lavigil, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, “LLaMa: Open and efficient foundation language models,” *arXiv preprint arXiv:2302.13971*, 2023.
- [25] I. Loshchilov, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2017.
- [26] P. Guo, X. Chang, S. Watanabe, and L. Xie, “Multi-speaker ASR combining non-autoregressive conformer CTC and conditional speaker chain,” in *Proc. Interspeech*, 2021, pp. 3720–3724.
- [27] D. Raj, D. Povey, and S. Khudanpur, “SURT 2.0: Advances in transducer-based multi-talker speech recognition,” *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 31, p. 3800–3813, 2023.
- [28] J. Cosentino, M. Pariente, S. Cornell, A. Deleforge, and E. Vincent, “Librimix: An open-source dataset for generalizable speech separation,” in *arXiv preprint arXiv:2005.11262*, 2020.
- [29] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaikos, W. Kraaij, M. Kronenthal *et al.*, “The AMI meeting corpus: A pre-announcement,” in *International workshop on machine learning for multimodal interaction*. Springer, 2005, pp. 28–39.
- [30] 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.
- [31] G. Wichern, J. Antognini, M. Flynn, L. R. Zhu, E. McQuinn, D. Crow, E. Manilow, and J. L. Roux, “Wham!: Extending speech separation to noisy environments,” in *Proc. Interspeech*, 2019, pp. 1368–1372.
- [32] X. Anguera, C. Wooters, and J. Hernando, “Acoustic beamforming for speaker diarization of meetings,” *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 15, no. 7, pp. 2011–2022, 2007.
- [33] D. Le, M. Jain, G. Keren, S. Kim, Y. Shi, J. Mahadeokar, J. Chan, Y. Shangguan, C. Fuegen, O. Kalinli *et al.*, “Contextualized streaming end-to-end speech recognition with trie-based deep biasing and shallow fusion,” *arXiv preprint arXiv:2104.02194*, 2021.
- [34] G. Sun, C. Zhang, and P. C. Woodland, “Tree-constrained pointer generator with graph neural network encodings for contextual speech recognition,” in *Proc. Interspeech*, 2022, pp. 2043–2047.