Interspeech 2024
1-5 September 2024, Kos, Greece

A Transcription Prompt-based Efficient Audio Large Language Model for Robust Speech Recognition

Yangze Li[1,2†], *Xiong Wang*[2†], *Songjun Cao*[2], *Yike Zhang*[2], *Long Ma*[2], *Lei Xie*[1*]

[1]Audio, Speech and Language Processing Group (ASLP@NPU),
Northwestern Polytechnical University, China
[2]Tencent YouTu Lab, China

yzli@mail.nwpu.edu.cn, chnxwang@tencent.com, lxie@nwpu.edu.cn

Abstract

Audio-LLM introduces audio modality into a large language model (LLM) to enable a powerful LLM to recognize, understand, and generate audio. However, during speech recognition in noisy environments, we observed the presence of illusions and repetition issues in audio-LLM, leading to substitution and insertion errors. This paper proposes a transcription prompt-based audio-LLM by introducing an ASR expert as a transcription tokenizer and a hybrid Autoregressive (AR) Non-autoregressive (NAR) decoding approach to solve the above problems. Experiments on 10k-hour WenetSpeech Mandarin corpus show that our approach decreases 12.2% and 9.6% CER relatively on Test_Net and Test_Meeting evaluation sets compared with baseline. Notably, we reduce the decoding repetition rate on the evaluation set to zero, showing that the decoding repetition problem has been solved fundamentally.

Index Terms: audio-LLM, speech recognition, hallucination of LLM, decoding repetition

1. Introduction

LLMs [1, 2, 3, 4] based on decoder-only Transformer [5] have revolutionized the field of natural language processing (NLP). Due to their ability to capture complex linguistic patterns and contextual information, LLMs perform impressive results on NLP tasks like machine translation, sentiment analysis, text generation, etc. Against this background, a significant amount of recent research has aimed at creating a seamless integration of text and audio through a unified large-scale audio-language model, enabling models to handle various tasks within and between these modalities.

Although unified audio models [6, 7, 8, 9, 10, 11, 12, 13] have shown considerable potential in tasks such as speech translation and speech understanding, their performance in speech recognition tasks still lacks robustness compared to well-tuned expert models, particularly in speech with complex acoustic environments. In this paper, we have observed the following issues introduced by the LLM-based framework have led to a degradation in the performance of audio-LLM speech recognition. The first issue is that the rich knowledge and associative abilities of LLM can lead to semantic corrections of recognition results, but may introduce substitution errors at the same time. The second one is the audio-LLM will lead to text fragment repetition during AR decoding in speech recognition tasks. This leads to many insertion errors and makes the recognition results difficult to comprehend. Although the above issues can be addressed in NLP tasks using common strategies such as temperature and top-p [14]/top-k [15], there is currently no effective solution for these problems in speech recognition due to the need for accurate transcriptions.

The main reason for the above issues is that prior works tend to introduce speech modality only through pre-trained ASR encoders but ignore information about the textual modality of speech, and the hallucination of LLM is not alleviated by a specific design for the speech recognition task. In this paper, inspired by such work like GER [16, 17, 18] which demonstrates that utilizing LLM for post-processing ASR transcriptions can also enhance recognition performance, we propose a transcription prompt-based audio-LLM that combines information from both the speech and the text modality obtained from ASR well-tuned expert models to enhance the speech recognition performance of the audio-LLM model. Our specific approaches to addressing these issues are as follows:

i) We propose an effective training framework that utilizes both modalities. Following the structure of Whipser [19], we concatenate the recognition transcriptions generated by an ASR expert model as textual prompts before the speech embedding and employ special token sequences to guide the task. Our approach enhances speech recognition performance effectively by helping LLM extract semantic information from speech modalities and improving their contextual modeling capabilities through the additional transcription prompts generated from a NAR ASR expert model trained with CTC (Connectionist Temporal Classification) [20] loss. This ASR expert model can constrain LLM from the textual modality to avoid the additional transcribe errors caused by its excessive generation ability.

ii) Since the CTC loss function establishes a time alignment between speech and text resulting in the problem of decoding repetition, we further propose a hybrid AR NAR decoding approach that uses textual prompts during the decoding step. This approach can solve the decoding repetition problem of audio-LLM fundamentally and achieves a lower ASR decoding real-time factor (RTF) by the hybrid approach.

Our proposed approach is mainly evaluated on 10k-hour WenetSpeech [21] Mandarin corpus. From the results, our model achieves superior performance on the Test_Net and Test_Meeting evaluation dataset, decreasing 12.2% and 9.6% on CER relatively compared with the baseline model, while significantly accelerating the decoding step with 32% relative time reduction. Furthermore, our results on AISHELL-1 [22] corpus indicate that our approach has strong generalization capabilities for low-cost domain adaptation. By analyzing the decoding repetition rates on each evaluation set, we reduce this rate to zero, showing that we completely solved this problem with our proposed hybrid AR NAR decoding approach.

† Equal contribution, * Corresponding author.

10.21437/Interspeech.2024-968

2. Related Work

Integration methods: Decoder-only LLMs can control and influence the generated text through prompt design, hence several multimodal models based on LLMs have been developed to expand the application of LLMs beyond text-based tasks. Models like speechGPT [6] and AudioPaLM [7] use discrete representations such as speech tokens to help LLMs finish speech-related tasks [13, 23], but these models may lose some speech-related information due to the conversion of continuous speech signals into discrete tokens. Experiments in LauraGPT [11] have shown that this conversion process causes a decline in performance on speech-related tasks compared to models that use continuous speech features. Besides, fine-tuning LLMs is difficult due to the large number of parameters for LLMs hence avoid doing this in most cases. The SLM [8] and Qwen-Audio [10] have shown that they achieve impressive performance on various speech-related tasks while keeping the LLM frozen even though the LLMs have been fine-tuned in LauraGPT [11] and Salmonn [9].

Repetition Problem: Text generation tasks in NLP usually use likelihood as a training objective to yield high-performance models. However, for decode-only models such as LLM, output text may become dull, incoherent, or stuck in repetitive loops while using maximization-based decoding approaches like greedy search. To avoid this problem, the current widely-used approach is to perform sampling strategies on the predicted probabilities during decoding, such as nucleus sampling [14] and top-k sampling [15]. However, for classification tasks like speech recognition, the decoding result from the model should be unique. Therefore, these probabilistic sampling strategies are not suitable for the ASR task because they may introduce additional errors, which are also verified by some experiments in this paper. In addition, the repetition problem is usually difficult to solve through some simple rules such as maximum decoding token limit, so our proposed approach gives valid guidance for solving the repetition problem in speech-related classification tasks of audio-LLM.

3. Proposed Method

3.1. Model architecture

As shown in Fig. 1, the audio-LLM in our approach consists of four main components: an LLM, a transcription tokenizer, a speech encoder, and an adapter.

LLM Audio-LLM is built upon an LLM, which serves as its fundamental component. Based on the powerful and flexible decoder-only structure of LLM, we can easily concatenate the input sequences of text and speech modalities, allowing LLM to learn the information between the two modalities autonomously.

Transcription tokenizer To further discover the powerful language capabilities of LLM, we introduce a tokenizer to provide the transcription prompt for input speech. In this paper, the tokenizer is an ASR pre-trained model using CTC loss. This tokenizer will decode input speech to text using CTC greedy search, then the text will be converted to discrete semantic representation by the text embedding layer of the LLM.

Speech encoder We employ the speech encoder with the same architecture and initialization as the transcription tokenizer without the project layer of output, and this component will be trained during the training stage. Ultimately, the encoder converts roughly every segment of the original audio signal into a high-dimensional representation.

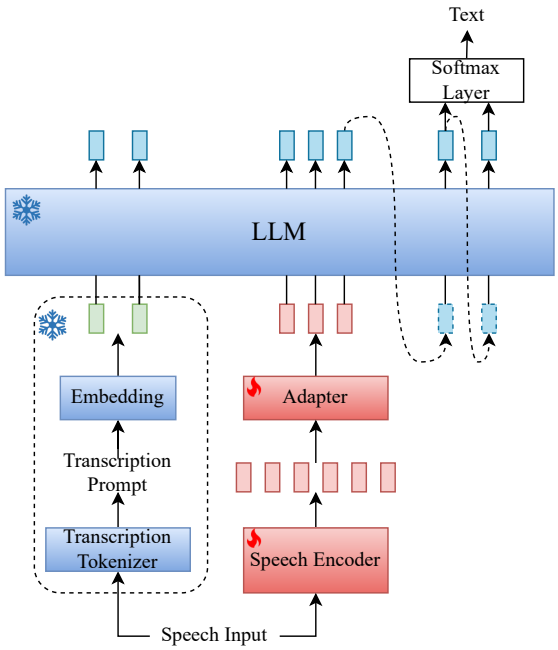

Figure 1: *The overview of our audio-LLM architecture.*

Adapter The adapter component connects the representations generated by the speech encoder with the text embeddings of the LLM, initializing with random weights during the training stage. This module contains such layers of 1D convolution and a fully connected layer, which maps the high-dimensional speech encoder output to the LLM text embedding. As a result, the adapter is optimized to map each segment of the input speech into the continuous semantic space of LLM.

3.2. Training framework

For speech recognition task of audio-LLM, training stage need paired data denoted as $(\mathbf{x}, \mathbf{y})$, where $\mathbf{x}$ represents speech input and $\mathbf{y}$ represents the corresponding text sequences $\{y_0, y_1, \cdots, y_{N-1}\}$. As shown in Eq.(1) and Eq.(2), the main objective during training is to maximize the probability of the next text token y_n given last token sequence $\mathbf{y}_{<n}$ and high-dimensional representation $\mathbf{H}_s$ generate by speech encoder and adapter, where $\mathcal{L}_{\text{CE}}$ is the loss function to optimize.

$$\mathbf{H}_s = \text{Adapter}(\text{Encoder}(\mathbf{x})) \tag{1}$$

$$\mathcal{L}_{\text{CE}} = -\sum_{n=0}^{N-1} \log \mathcal{P}_{LLM}\left(y_n \mid \mathbf{y}_{<n}, \mathbf{H}_s; \Theta_{\text{LLM}}\right) \tag{2}$$

By taking the transcription generated by tokenizer into the prompt, the loss function $\mathcal{L}_{\text{CE\_prompt}}$ is described as shown in Eq.(3), where transcription prompt $\mathbf{y}_{\text{prompt}} = \text{Tokenizer}(\mathbf{x})$.

$$\mathcal{L}_{\text{CE\_prompt}} = -\sum_{n=0}^{N-1} \log \mathcal{P}_{LLM}\left(y_n \mid \mathbf{y}_{<n}, \mathbf{y}_{\text{prompt}}, \mathbf{H}_s; \Theta_{\text{LLM}}\right) \tag{3}$$

During the training stage, to avoid the model's over-fitting on the transcription prompt, we use a hyper-parameter $\lambda \in [0, 1]$ to control whether the current utterance has a transcription prompt or not. As a result, the training loss function for each utterance

in a training batch is shown in Eq.(4),

$$\mathcal{L} = \begin{cases} \mathcal{L}_{\text{CE\_prompt}} & , p \leqslant \lambda \\ \mathcal{L}_{\text{CE}} & , p > \lambda \end{cases} \tag{4}$$

while p is a random number generated for each utterance in every training batch with a uniform distribution in [0, 1].

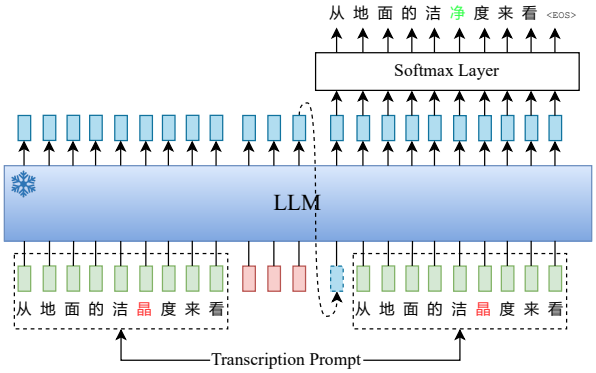

Figure 2: *NAR decoding approach combined with transcription prompt.*

3.3. Decoding

AR decoding Audio-LLM usually uses the AR decoding approach, which predicts the next token by the last predicted token until $<$ EOS $>$ is predicted as shown in Eq.(5). In this paper, the AR decoding approach is used as default if the experiment details do not mention the type of decoding approach.

$$y_n^* = \underset{y_n}{\text{argmax}}\ \mathbf{LLM}(\mathbf{y}_{<n}, \mathbf{H}_s; \Theta_{\text{LLM}}) \tag{5}$$

NAR decoding By introducing the transcription prompt decoded by the tokenizer, we propose a fast NAR audio-LLM decoding approach. As shown in Figure 2, we replace the context $\mathbf{y}_{<n}$ predicted by the LLM with $y_{\text{prompt}<n}$ predicted by the tokenizer. Finally, we can perform a NAR decoding approach that generates predicted text sequence $\mathbf{y}^*$ in one step described in Eq.(6).

$$\mathbf{y}^* = \underset{\mathbf{y}}{\text{argmax}}\ \mathbf{LLM}(\mathbf{y}_{\text{prompt}}, \mathbf{H}_s; \Theta_{\text{LLM}}) \tag{6}$$

In the NAR decoding approach, it can be described that the LLM modifies the transcription prompt and plays as an error correction model. Since the length of the predicted text sequence only relies on the length of the transcription prompt, this approach will avoid the problem of repetition.

Hybrid AR NAR decoding Although the NAR decoding approach can solve the repetition problem, the ability of the LLM is limited by the fixed length of the transcription prompt. To combine the advantages of AR and NAR decoding approaches, we propose a hybrid AR NAR decoding approach as shown in the algorithm 1. This hybrid approach determines whether there is a problem such as repetition in the AR decoding approach using decode length limit hyper-parameterσ, and then uses the NAR decoding result if the condition is triggered, to take into account the advantages of both AR and NAR decoding approaches. In this paper, the hyper-parameter σ is empirically set to 1.5.

Algorithm 1 Pipeline of Hybrid AR NAR decoding approach

Given a well-trained proposed audio-LLM
Given input speech feature $\mathbf{x}$ and decode parameter σ
Compute high-dimensional representation $\mathbf{H}_s$ from Eq.(1)
Compute transcription prompt $\mathbf{y}_{\text{prompt}} = \text{Tokenizer}(\mathbf{x})$, and get token number of $\mathbf{y}_{\text{prompt}}$ as L_{prompt}
Initialize the decode result $\mathbf{y}^*$ with an empty sequence, set length of decode result L_{decode} as 0
while $\mathbf{y}^*$ is not end with $<$ EOS $>$ **do**
  Generate next token y_n^* from Eq.(5)
  $L_{\text{decode}} = L_{\text{decode}} + 1$
  if $L_{\text{decode}} > \sigma \times L_{\text{prompt}}$ **then**
    Replace $\mathbf{y}^*$ with Eq.(6)
    return $\mathbf{y}^*$
  end if
  Append y_n^* to the end of $\mathbf{y}^*$
end while
return $\mathbf{y}^*$

4. Experiments

4.1. Dataset

In this paper, we evaluate our proposed approach to the WenetSpeech corpus [21]. This corpus contains over 10,000 hours of high-quality labeled Mandarin speech, which is sourced from YouTube and podcasts, covering different speaking styles, scenarios, domains, topics, and noise environments. We use two carefully checked evaluation sets Test_Net and Test_Meeting, the first one is a match set compared with training data, and the second one is a mismatch set that contains far-field and conversational meeting speech. In addition, we use the well-known public set AISHELL-1 [22] to confirm the out-of-domain performance of the model.

4.2. Experimental setup

In this paper, the LLM is initialized with pre-trained weights obtained from Qwen-7B [4]. Qwen-7B is a Transformer decoder model with 32 layers and a hidden size of 4096, comprising a total of 7.7 billion parameters. We implement a two-stage training approach for our proposed transcription prompt-based LLM. For the first stage, we trained a transcription tokenizer which used a learning rate of 1e-3 with a batch size of 256, performed 5000 steps of warm-up, and employed gradient accumulation with a factor of 16. The transcription tokenizer consists of 12 layers of Conformer [24] layers with a hidden size of 512. Besides, the transcription tokenizer takes in the 80-dimensional mel-filterbank feature with a 10 ms window shift and a 25 ms frame length. For the second stage, we initialize the speech encode with the same parameters of the transcription tokenizer trained in the first stage, then we used a learning rate of 1e-4 with a batch size of 64, performed a warm-up for 2000 steps, and employed gradient accumulation with a factor of 16. Besides, for the second stage, only the speech encoder and adapter are trained while the transcription tokenizer and LLM are kept frozen. This adapter contains 2 1D-convolution layers and 1 fully connected layer, which maps the dimension of the speech encoder from 512 to 4096. In addition, the first convolution layer uses a stride of two to do down-sampling. The trainable parameters of the speech encoder amount to 70 million, of which the adapter is 10 million. When using AR greedy decoding, we set the maximum decoding token limit to 200 (as

the evaluation set contained sentences with a maximum of 180 tokens).

Table 1: *CER (%) of various models on Test_Net, Test_Meeting and Test_aishell1. The RTF is computed as the ratio of the total inference time to the total duration of evaluation sets.*

Model	CER (%)			RTF
	Test_Net	Test_Meeting	Test_aishell1	
Baselines				
Conformer-W1	8.60	14.30	4.61	
Qwen-Audio	9.62	9.05	1.59	
Ours				
Audio-LLM				
+ $\lambda = 0.0$	9.18	15.30	4.11	
+ $\lambda = 0.5$	8.47	13.94	3.86	0.39
+NAR	8.35	14.62	3.83	**0.04**
+Hybrid AR NAR	**8.09**	13.83	**3.71**	0.25
+ $\lambda = 1.0$				
+Hybrid AR NAR	8.26	**13.65**	3.73	

Table 2: *Comparison of insertion, deletion, and substitution errors among different approaches on Test_Net.*

Model	Insertion	Deletion	Substitution
Audio-LLM			
+ $\lambda = 0.0$	4160	16408	25503
+ $\lambda = 0.5$	3732	12058	19291
+ NAR	1897	12035	20662
+ Hybrid AR NAR	2251	12258	19004

4.3. Analysis on transcription prompt

To analyze the effect of the transcription prompt for audio-LLM, we first set two baseline models Conformer-W1 and Qwen-Audio. Conformer-W1 is trained on WenetSpeech in the same setup as the CTC-based transcription tokenizer mentioned in Sec.4.2 and Qwen-Audio is an audio-LLM that has achieved a state-of-the-art speech recognition performance recently. The result of our proposed transcription prompt-based audio-LLM is shown in Tab. 1. From the result, for $\lambda = 0$ which means do not use transcription prompt during the training stage, the audio-LLM and Qwen-Audio get a worse result on Test_Net due to the more insertion errors caused by the hallucination of LLM. After we introduced the transcription prompt with $\lambda = 0.5$, the model got a significant improvement on evaluation sets compared with the model on $\lambda = 0$. Furthermore, we also compared the effect of different decoding approaches and resulting in our proposed hybrid AR NAR decoding approach will bring additional improvement for audio-LLM during the decoding stage, even if the CER is significantly lower than Qwen-Audio on Test_Net (9.62 $\rightarrow$ 8.09) and Conformer-W1 on Test_aishell1 (4.61 $\rightarrow$ 3.71). In addition, we designed an ablation experiment with $\lambda = 1.0$ to prove that over-reliance on the transcription prompt can take disadvantages to the audio-LLM.

To further analyze the detailed effects of the transcription prompt, we list the insertion, deletion, and substitution errors in Tab. 2. It shows that the model with $\lambda = 0.5$ has less error on the three types than the model with $\lambda = 0.0$, and after using the NAR decoding approach, insertion errors decrease a lot. This shows the transcription prompt can restrain its over-generation ability for the speech recognition task. Furthermore, the proposed hybrid AR NAR decoding approach will further decrease substitution errors which shows CTC transcription prompt can improve the modal alignment ability of audio-LLM. It is worth mentioning that the hybrid AR NAR decoding approach allows for earlier truncation of AR decoding when repetition problems arise, resulting in lower RTF compared to the AR decoding approach.

4.4. Analysis on repetition problem

We defined sentence-level decoding repetition ratio (DRR) as the number of sentences that fall into the repetition problem divided by the total number of the evaluation set, to measure the seriousness of the repetition problem. As shown in Tab. 3, after the introduction of the transcription prompt and the hybrid AR NAR decoding approach, the DRR will be reduced to 0 step by step, which means the repetition problem is completely solved. Compared with the existing approaches, we show the result of the top-3 samples strategy. As a result, the decoding problem repetition seems to be effectively alleviated, but resulting in an unacceptable CER increase, mainly because the ASR task is a classification task rather than a generative task.

Table 3: *The sentence-level decoding repetition ratio (DRR (‰)) for each model.*

Model	Test_Net		Test_Meeting	
	DRR	CER	DRR	CER
Qwen-Audio	0.43	9.62	0.83	9.05
Audio-LLM				
+ $\lambda = 0.0$	0.36	9.18	0.96	15.30
+ top-3 sample	0.03	11.14	0.60	17.64
+ $\lambda = 0.5$	0.28	8.47	0	13.94
+ AR NAR	0	8.09	0	13.83

Table 4: *CER (%) of different models on Test_Net and AISHELL-1, when $\lambda = 0.5$ for audio-LLM and hybrid AR NAR decoding approach is used.*

Model	Tokenizer	Test_Net	Test_aishell1
Conformer-W1	-	8.60	4.61
Conformer-A1	-	50.13	5.20
Audio-LLM-W1	Conformer-W1	8.09	3.71
Audio-LLM-W1	Conformer-A1	12.56	**3.08**

4.5. Generalization for robustness

To further evaluate how the transcription prompt affects the audio-LLM, we provided transcription prompts generated by another tokenizer different from the one during training. As shown in Tab. 4, audio-LLM-W1 means the model uses WenetSpeech to train and initialize the tokenizer and speech encoder, and Conformer-A1 means the ASR expert model trained with AISHELL-1 corpus. The results show different tokenizers will lead the audio-LLM to related domains, proving our proposed approach has the robustness to achieve domain adaptation.

5. Conclusion

In this paper, we proposed a transcription prompt-based audio-LLM for the ASR task. Specifically, we introduced a transcription tokenizer to generate transcription prompts for audio-LLM and a hybrid AR NAR decoding approach to avoid the hallucination and repetition problems of audio-LLM for the ASR task. As a result, our proposed approach evaluated on WenetSpeech can decrease 12.2% and 9.6% on CER relatively on the Test_Net and Test_Meeting compared with the baseline model, and we reduce the sentence-level decoding repetition ratio to zero, resulting we completely solved the repetition problem. Besides, our generalization validation experiment also shows our proposed method has the ability of low-cost domain adaptation for audio-LLM.

6. References

[1] OpenAI, "GPT-4 technical report," *CoRR*, vol. abs/2303.08774, 2023.

[2] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," *CoRR*, vol. abs/2302.13971, 2023.

[3] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif *et al.*, "Palm: Scaling language modeling with pathways," *J. Mach. Learn. Res.*, vol. 24, pp. 240:1–240:113, 2023.

[4] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, B. Hui, L. Ji, M. Li, J. Lin, R. Lin, D. Liu, G. Liu, C. Lu, K. Lu, J. Ma *et al.*, "Qwen technical report," *CoRR*, vol. abs/2309.16609, 2023.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Annual Conference on Neural Information Processing Systems, NeurIPS 2017*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 5998–6008.

[6] D. Zhang, S. Li, X. Zhang, J. Zhan, P. Wang, Y. Zhou, and X. Qiu, "Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities," in *Conference on Empirical Methods in Natural Language Processing, EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Association for Computational Linguistics, 2023, pp. 15 757–15 773.

[7] P. K. Rubenstein, C. Asawaroengchai, D. D. Nguyen, A. Bapna, Z. Borsos, F. de Chaumont Quitry, P. Chen, D. E. Badawy, W. Han, E. Kharitonov, H. Muckenhirn, D. Padfield, J. Qin, D. Rozenberg, T. N. Sainath, J. Schalkwyk, M. Sharifi, M. T. Ramanovich, M. Tagliasacchi, A. Tudor *et al.*, "Audiopalm: A large language model that can speak and listen," *CoRR*, vol. abs/2306.12925, 2023.

[8] M. Wang, W. Han, I. Shafran, Z. Wu, C. Chiu, Y. Cao, N. Chen, Y. Zhang, H. Soltau, P. K. Rubenstein, L. Zilka, D. Yu, G. Pundak, N. Siddhartha, J. Schalkwyk, and Y. Wu, "SLM: bridge the thin gap between speech and text foundation models," in *Automatic Speech Recognition and Understanding Workshop, ASRU 2023*. IEEE, 2023, pp. 1–8.

[9] C. Tang, W. Yu, G. Sun, X. Chen, T. Tan, W. Li, L. Lu, Z. Ma, and C. Zhang, "SALMONN: towards generic hearing abilities for large language models," *CoRR*, vol. abs/2310.13289, 2023.

[10] Y. Chu, J. Xu, X. Zhou, Q. Yang, S. Zhang, Z. Yan, C. Zhou, and J. Zhou, "Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models," *CoRR*, vol. abs/2311.07919, 2023.

[11] J. Wang, Z. Du, Q. Chen, Y. Chu, Z. Gao, Z. Li, K. Hu, X. Zhou, J. Xu, Z. Ma, W. Wang, S. Zheng, C. Zhou, Z. Yan, and S. Zhang, "Lauragpt: Listen, attend, understand, and regenerate audio with GPT," *CoRR*, vol. abs/2310.04673, 2023.

[12] Y. Shen, K. Song, X. Tan, D. Li, W. Lu, and Y. Zhuang, "Hugginggpt: Solving AI tasks with chatgpt and its friends in huggingface," *CoRR*, vol. abs/2303.17580, 2023.

[13] T. Wang, L. Zhou, Z. Zhang, Y. Wu, S. Liu, Y. Gaur, Z. Chen, J. Li, and F. Wei, "Viola: Unified codec language models for speech recognition, synthesis, and translation," *CoRR*, vol. abs/2305.16107, 2023.

[14] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," in *International Conference on Learning Representations, ICLR 2020*. OpenReview.net, 2020.

[15] A. Fan, M. Lewis, and Y. N. Dauphin, "Hierarchical neural story generation," in *Annual Meeting of the Association for Computational Linguistics, ACL 2018*, I. Gurevych and Y. Miyao, Eds. Association for Computational Linguistics, 2018, pp. 889–898.

[16] C. H. Yang, Y. Gu, Y. Liu, S. Ghosh, I. Bulyko, and A. Stolcke, "Generative speech recognition error correction with large language models and task-activating prompting," in *Automatic Speech Recognition and Understanding Workshop, ASRU 2023*. IEEE, 2023, pp. 1–8.

[17] C. Chen, Y. Hu, C. H. Yang, S. M. Siniscalchi, P. Chen, and C. E. Siong, "Hyporadise: An open baseline for generative speech recognition with large language models," in *Annual Conference on Neural Information Processing Systems, NeurIPS 2023*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., 2023.

[18] Y. Hu, C. Chen, C.-H. H. Yang, R. Li, C. Zhang, P.-Y. Chen, and E. S. Chng, "Large language models are efficient learners of noise-robust speech recognition," in *International Conference on Learning Representations, ICLR 2024*. OpenReview.net, 2024.

[19] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International Conference on Machine Learning, ICML 2023*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 2023, pp. 28 492–28 518.

[20] A. Graves, S. Fernández, F. J. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *International Conference on Machine Learning, ICML 2006*, ser. ACM International Conference Proceeding Series, W. W. Cohen and A. W. Moore, Eds., vol. 148. ACM, 2006, pp. 369–376.

[21] B. Zhang, H. Lv, P. Guo, Q. Shao, C. Yang, L. Xie, X. Xu, H. Bu, X. Chen, C. Zeng, D. Wu, and Z. Peng, "WENETSPEECH: A 10000+ hours multi-domain mandarin corpus for speech recognition," in *International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022*. IEEE, 2022, pp. 6182–6186.

[22] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, "AISHELL-1: an open-source mandarin speech corpus and a speech recognition baseline," in *Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment, O-COCOSDA 2017*. IEEE, 2017, pp. 1–5.

[23] Q. Chen, W. Wang, Q. Zhang, S. Zheng, S. Zhang, C. Deng, Y. Ma, H. Yu, J. Liu, and C. Zhang, "Loss masking is not needed in decoder-only transformer for discrete-token based ASR," *CoRR*, vol. abs/2311.04534, 2023.

[24] A. Gulati, J. Qin, C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Annual Conference of the International Speech Communication Association, INTERSPEECH 2020*, H. Meng, B. Xu, and T. F. Zheng, Eds. ISCA, 2020, pp. 5036–5040.