Unlocking Large Audio-Language Models for Interactive Language Learning

Anonymous ACL submission

Abstract

Achieving pronunciation proficiency in a second language (L2) remains a challenge, despite the development of Computer-Assisted Pronunciation Training (CAPT) systems. Traditional CAPT systems often provide unintuitive feedback that lacks actionable guidance, limiting its effectiveness. Recent advancements in audio-language models (ALMs) offer the potential to enhance these systems by providing more user-friendly feedback. In this work, we investigate ALMs for chat-based pronunciation training by introducing L2-Arctic-plus, a dataset with detailed error explanations and actionable suggestions for improvement. We benchmark cascaded ASR+LLMs and existing ALMs on this dataset, specifically in detecting mispronunciation and generating actionable feedback. To improve the performances, we further propose to instruction-tune ALMs on L2-Arctic-plus. Experimental results demonstrate that our instruction-tuned models significantly outperform existing baselines on mispronunciation detection and suggestion generation in terms of both objective and human evaluation, highlighting the value of the proposed dataset.

1 Introduction

The acquisition of a second language (L2) is a fundamental necessity in bilingual and multilingual communities. However, attaining a high level of proficiency in pronunciation and language usage remains a considerable challenge for L2 learners. Computer-Assisted Pronunciation Training (CAPT) systems have been developed as effective tools to support L2 learners by detecting, diagnosing, and assessing mispronunciations (Eskenazi, 2009; Rogerson-Revell, 2021). Conventional CAPT systems primarily focus on providing detailed feedback at the phoneme, word, and utterance levels for mispronunciation detection and fluency evaluation (Witt and Young, 2000; Zhang et al., 2021; Kheir et al., 2023), thereby facilitating

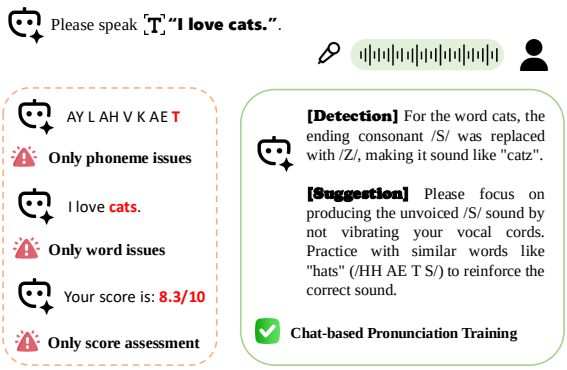

Figure 1: Illustrative examples of chat-based pronunciation training for interactive language learning. The system generates detection with error explanations and suggestions with practical corrective actions to provide more user-friendly feedback.

targeted practice and enabling learners to enhance their language skills through systematic error correction.

Despite significant achievements in developing robust models for mispronunciation detection and pronunciation assessment, existing methods primarily provide location-based diagnostic feedback (Xu et al., 2021) and score-based assessment feedback (Gong et al., 2022). However, such feedback is often unintuitive and challenging for L2 learners to interpret, particularly in terms of actionable suggestions for improvement. Recent advances in large-scale speech-language models and audio-language models (ALMs) have demonstrated remarkable performance across various speech and audio-related tasks, including automatic speech recognition (ASR), speech synthesis, and spoken dialogue systems (Chu et al., 2023, 2024; Zhang et al., 2023; Huang et al., 2024; Deshmukh et al., 2023). Nevertheless, their application in interactive language learning, particularly for the complex task of chat-based pronunciation training, remains largely unexplored. The integration of language models presents an opportunity to enhance acoustic analysis by providing user-friendly feedback,

such as text-based explanations of pronunciation errors along with actionable suggestions for improvement, as shown in Figure 1.

In this work, we investigate the potential of large ALMs as language instructors to enhance language learning, with a particular emphasis on *chat-based pronunciation training*. Our goal is to provide interpretable, text-based feedback that includes detailed error explanations and actionable suggestions. To facilitate this task, we introduce **L2-Arctic-plus**, an extension of the L2-Arctic dataset (Zhao et al., 2018), which incorporates text-based annotations for error explanations and actionable suggestions. Furthermore, we examine the application of the cascaded ASR+LLM framework for chat-based pronunciation training. Our analysis reveals that ASR models often rectify pronunciation errors in the input, yielding an accurate transcription for LLMs and thereby limiting LLMs' ability to detect pronunciation errors from the original audio. Additionally, our evaluation of existing large ALMs on this task indicates their significant limitations in both accurate mispronunciation detection and actionable feedback generation. As a consequence, we propose to improve chat-based pronunciation training by instruction-tuning ALMs using the L2-Arctic-plus training set. Experimental results demonstrate that our instruction-tuned ALM outperforms existing baselines, achieving substantial improvements in chat-based pronunciation training.

Our key contributions are summarized below:

- We construct L2-Arctic-plus, a novel benchmark designed for chat-based pronunciation training in interactive language learning. This dataset is specifically developed for audio-language models and includes text-based annotations on pronunciation error explanations and actionable corrective suggestions.

- We systematically analyze the performance of ASR+LLM cascades and existing ALMs in chat-based pronunciation training. We further improve this novel task by instruction-tuning the ALMs on a curated training set of L2-Arctic-plus, demonstrating significant improvements in both mispronunciation detection and feedback generation.

- This work expands the capability scope of ALMs in the domain of chat-based pronunciation training, addressing an important gap in language learning.

2 Related Work

Audio-Language Modeling. The development of multimodal large language models has recently expanded beyond vision-based modalities to include audio and video, leading to increased research interest in audio-language models. Prominent models such as Qwen-Audio (Chu et al., 2023), Qwen2-Audio (Chu et al., 2024), SpeechGPT (Zhang et al., 2023), Audio-GPT (Huang et al., 2024), Pengi (Deshmukh et al., 2023), and GPT-4o (Hurst et al., 2024) demonstrate remarkable versatility, addressing a wide array of downstream tasks, including speech, sound, and music processing. These efforts seek to unify diverse audio-related tasks within a single foundation model. Despite their impressive capabilities, these models have limited applications in pronunciation detection, a critical task in language learning. Notably, prior acoustic models have demonstrated effectiveness in pronunciation detection tasks (Hu et al., 2015; Xu et al., 2021; Korzekwa et al., 2021), highlighting the gap in current audio-language models for educational applications.

Computer-Assisted Pronunciation Training. CAPT has become an essential component of modern language learning, leveraging technological advancements to enhance learners' pronunciation proficiency. Early CAPT systems primarily relied on repetitive drills and rudimentary feedback mechanisms, utilizing basic audio playback and recording features (Amrate and Tsai, 2024). The introduction of ASR technology has enabled more interactive and adaptive training environments, facilitating real-time feedback on pronunciation (Arora et al., 2018; Henrichsen, 2021). More recently, CAPT systems have further integrated machine learning to deliver more sophisticated feedback, encompassing the evaluation of prosodic features such as intonation, stress, and rhythm (Eskenazi, 2009; Rogerson-Revell, 2021). Contemporary CAPT methodologies emphasize detailed assessments at the phoneme, word, and utterance levels (Gong et al., 2022; Kheir et al., 2023), enabling learners to accurately distinguish and produce specific consonants and vowels while addressing suprasegmental features like stress patterns, intonation, and rhythm. However, existing CAPT approaches often lack comprehensive and interpretable feedback, underscoring the need for further advancements to enhance the effectiveness of pronunciation training systems.

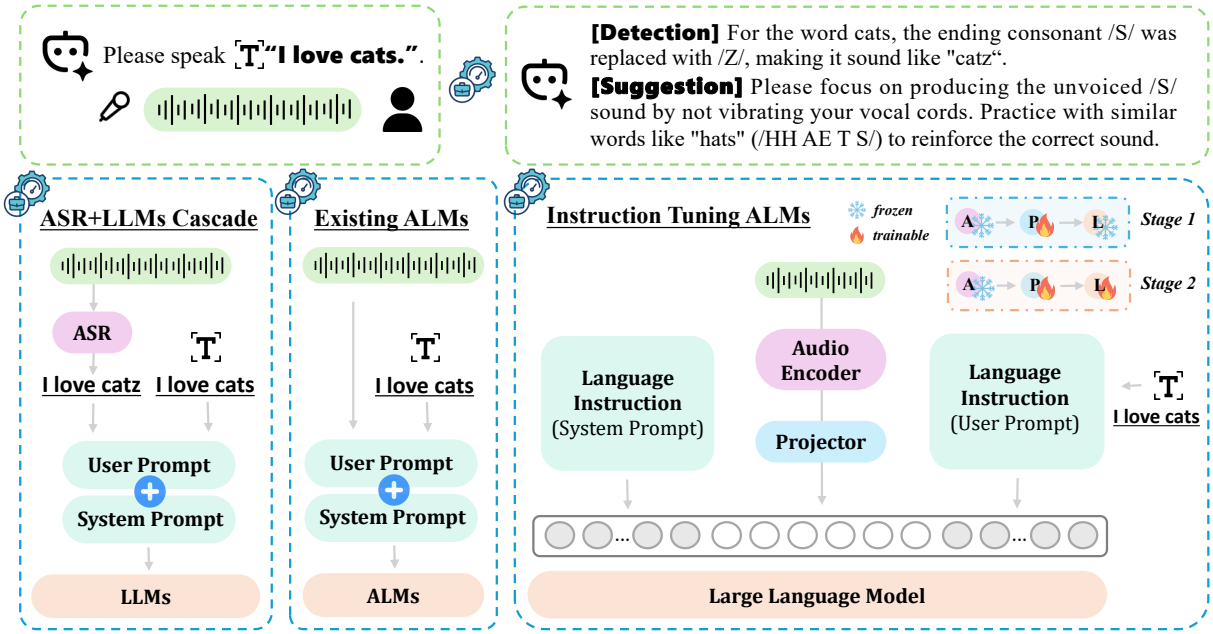

Figure 2: Overview of (*left*) ASR+LLMs cascade; (*middle*) existing ALMs; (*right*) instruction-tuning ALMs. For instruction-tuning ALMs, the upper right corner shows the trainable module in a two-stage pipeline. A represents the Audio Encoder, P represents the Projector, and L represents the Large Language Model.

3 Interactive Language Learning

3.1 Problem Statement

This study focuses on *chat-based pronunciation training* within the context of interactive language learning. In this framework, the user is instructed to read a canonical text sequence, denoted as $\boldsymbol{W}_{1:N}$, where N represents the total number of words. The user's speech is then recorded as an audio sample, $\boldsymbol{X}_A$. The primary objective of the chat-based pronunciation training system, denoted as $f_\theta(\cdot)$, where θ represents model parameters, is to generate text-based responses: $\boldsymbol{Y}_R = f_\theta(\boldsymbol{X}_A)$. This response is designed to identify mispronunciation in the user's speech and provide corresponding actionable suggestions for improvement through an interactive chat-based interface.

3.2 Dataset Curation of L2-Arctic-plus

Since no existing datasets are specifically designed for chat-based pronunciation training, especially without ground-truth responses $\boldsymbol{Y}_R$, we introduce **L2-Arctic-plus** as a benchmark for this task. L2-Arctic-plus is built upon the L2-Arctic dataset (Zhao et al., 2018), a non-native English corpus designed for mispronunciation detection with frame-level annotations. The original L2-Arctic dataset consists of speech recordings from 24 non-native English speakers (12 males, 12 females) with diverse native languages including Hindi, Korean, Mandarin, Spanish, Arabic, and Vietnamese.

Following prior practices in Peng et al. (2021); Feng et al. (2020); Yang et al. (2022), we select the same 900 samples as the evaluation set. Each sample comprises a speech recording $\boldsymbol{X}_A$ along with manual annotations, including canonical word sequences $\{\boldsymbol{W}_n\}_{n=1}^N$, a binary mispronunciation indicator $\boldsymbol{D} \in \{0, 1\}$ – where $\boldsymbol{D}(\boldsymbol{W}_n) = 1$ denotes that the n-th word $\boldsymbol{W}_n$ is mispronounced – and a mispronunciation type indicator $\boldsymbol{E} \subseteq \{S, D, I\}$. Here, $\boldsymbol{E}(\boldsymbol{W}_n)$ represents the set of mispronunciation types (Substitution, Deletion, or Insertion) present in the n-th word $\boldsymbol{W}_n$, with $\boldsymbol{D}(\boldsymbol{W}_n) = 0$ if no mispronunciation is detected $\boldsymbol{E}(\boldsymbol{W}_n) = \emptyset$. The annotations are based on phonemes, so a single word may contain multiple phonemic errors which may belong to different types. In these annotations, the mispronounced phonemes and their corresponding error types are clearly marked. Based on these existing annotations, we illustrate how to construct new ground-truth responses $\boldsymbol{Y}_R$ following a coarse-to-fine manner through a two-stage process.

In the first stage, we generate initial responses by formulating a structured prompt and utilizing the existing annotations as input to query GPT-4o (Hurst et al., 2024). The model generates feedback that includes both mispronunciation error explanations and corrective suggestions. An example of the prompt-response interaction is illustrated in Appendix Figure 3 and Figure 4. Specifically, the response is structured as a se-

quence of word-level error-suggestion pairs $\boldsymbol{Y}_R = \{\boldsymbol{W}^{(l)}\colon [\boldsymbol{Y}_E{}^{(l)}, \boldsymbol{Y}_S{}^{(l)}]\}_{l=1}^{L}$, where $\boldsymbol{W}^{(l)}$ represents the l-th mispronunced word $\boldsymbol{D}(\boldsymbol{W}^{(l)}) = 1$, $\boldsymbol{Y}_E{}^{(l)}$ refers to a text-based explanation of the mispronunciation type and $\boldsymbol{Y}_S{}^{(l)}$ represents a corrective suggestion on how to improve the pronunciation given this error explanation $\boldsymbol{Y}_E{}^{(l)}$. The total number of pairs, L, corresponds to the total number of mispronounced words $L = \sum_{n=1}^{N} \boldsymbol{D}(\boldsymbol{W}_n)$.

In the second stage, three human annotators are involved to verify GPT-4o-generated responses in terms of the correctness of both error explanation and corrective suggestion $[\boldsymbol{Y}_E{}^{(l)}, \boldsymbol{Y}_S{}^{(l)}]$. If any responses contain incorrect explanations or inappropriate suggestions, we prompt GPT-4o to regenerate new responses, followed by another round of human verification. The final verified responses constitute the ground-truth annotations in L2-Arctic-plus.

3.3 Evaluation Protocols

This subsection outlines the evaluation protocols for assessing a chat-based pronunciation training system $f_\theta(\cdot)$ on the L2-Arctic-plus dataset. Given a generated response $\hat{\boldsymbol{Y}}_R = \{\hat{\boldsymbol{W}}^{(l)}\colon [\hat{\boldsymbol{Y}}_E^{(l)}, \hat{\boldsymbol{Y}}_S^{(l)}]\}_{l=1}^{\hat{L}}$ and a reference response $\boldsymbol{Y}_R$, the evaluation consists of both objective and subjective assessments. Objective evaluation measures performance in mispronunciation detection and feedback generation, while subjective evaluation involves human judgment.

Mispronunciation Detection Evaluation. To evaluate mispronunciation detection, we compute standard classification metrics: True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN). Unlike prior acoustic-only approaches with frame-level evaluation (Xu et al., 2021), our framework adopts a **word-level** evaluation scheme:

$$\text{TP} = \sum_{n=1}^{N} \mathbb{I}(\boldsymbol{D}(\hat{\boldsymbol{W}}_n) = 1 \wedge \boldsymbol{D}(\boldsymbol{W}_n) = 1); \quad (1)$$

$$\text{FP} = \sum_{n=1}^{N} \mathbb{I}(\boldsymbol{D}(\hat{\boldsymbol{W}}_n) = 1 \wedge \boldsymbol{D}(\boldsymbol{W}_n) = 0); \quad (2)$$

$$\text{FN} = \sum_{n=1}^{N} \mathbb{I}(\boldsymbol{D}(\hat{\boldsymbol{W}}_n) = 0 \wedge \boldsymbol{D}(\boldsymbol{W}_n) = 1); \quad (3)$$

$$\text{TN} = \sum_{n=1}^{N} \mathbb{I}(\boldsymbol{D}(\hat{\boldsymbol{W}}_n) = 0 \wedge \boldsymbol{D}(\boldsymbol{W}_n) = 0). \quad (4)$$

We report Precision, Recall, and F1-score, computed across all samples rather than averaging per entry. Additionally, we introduce the *Extra Words Ratio* (**EWR**) to evaluate the system's tendency to introduce spurious words absent from the canonical text $\boldsymbol{W}_{1:N}$. Specifically, EWR is defined as follows:

$$\text{EWR} = \frac{1}{M} \sum_{j=1}^{M} \mathbb{I}(\hat{\boldsymbol{W}}_j \notin \{\boldsymbol{W}_n\}_{n=1}^{N}), \quad (5)$$

where M is the total number of words predicted by the system. A higher EWR indicates a greater tendency to hallucinate non-existent words, reflecting lower system reliability in mispronunciation detection.

Feedback Generation Evaluation. To assess the quality of generated feedback, we compare the system-generated error-suggestion pairs $\{\hat{\boldsymbol{W}}^{(l)}\colon [\hat{\boldsymbol{Y}}_E^{(l)}, \hat{\boldsymbol{Y}}_S^{(l)}]\}_{l=1}^{\hat{L}}$ against the referenced ground-truth pairs $\{\boldsymbol{W}^{(l)}\colon [\boldsymbol{Y}_E{}^{(l)}, \boldsymbol{Y}_S{}^{(l)}]\}_{l=1}^{L}\}$. For objective evaluations, we calculate metrics: **BLEU-2** (Papineni et al., 2002), measuring 2-gram overlap between system outputs and ground truth; **ROUGE-L** (Lin, 2004), measuring the longest common subsequence; and **BERTScore** (Zhang et al., 2019), calculating semantic similarity leveraging contextual embeddings. Additionally, we conduct subjective human evaluations to assess the suggestion relevance, interpretability, and helpfulness of the generated feedback.

4 Investigating ASR+LLMs Cascade

LLMs have been increasingly integrated into speech-related tasks such as ASR (Ma et al., 2024; Geng et al., 2024). Since LLMs can not directly process audio input, a common approach is to employ a pre-trained ASR model to transcribe speech into text, enabling LLMs to handle downstream tasks. This section explores the potential of the ASR+LLMs cascade for chat-based pronunciation training, evaluating its effectiveness in mispronunciation detection and suggestion generation.

4.1 Cascaded ASR+LLM Framework

ASR-based Transcription. ASR models serve as the foundational component for speech-to-text transcription. In this framework, we utilize the pre-trained ASR model to transcribe the given speech recordings $\boldsymbol{X}_A$ into text $\hat{\boldsymbol{W}}_{1:\hat{N}}$. We assume that mispronounced words would be transcribed into

ASR Models	LLMs	Mispronunciation Detection				Suggestion Generation		
		Precision ↑	Recall ↑	F1 ↑	EWR ↓	BLEU-2 ↑	ROUGE-L ↑	BERTScore ↑
Whisper Small	Mistral-7B	**53.6**	**4.9**	**9.0**	0.3	4.5	7.0	**79.8**
Whisper Medium	Mistral-7B	48.2	4.0	7.4	0.3	**4.6**	**7.1**	**79.8**
Whisper Large	Mistral-7B	48.9	3.4	6.4	**0.1**	4.1	6.1	79.5
Wav2vec2 Base	Mistral-7B	**52.8**	**6.8**	**12.1**	0.4	**5.0**	**8.5**	**80.5**
Wav2vec2 Large	Mistral-7B	51.2	4.5	8.3	**0.3**	4.7	7.2	79.9
Whisper Small	Llama-3.1-8B	**53.3**	**12.1**	**19.7**	0.9	**6.6**	**12.8**	**82.1**
Whisper Medium	Llama-3.1-8B	51.9	10.2	17.0	1.0	5.8	11.5	81.7
Whisper Large	Llama-3.1-8B	52.8	8.4	14.5	**0.7**	5.5	10.7	81.4
Wav2vec2 Base	Llama-3.1-8B	53.8	**17.8**	**26.8**	1.1	**7.3**	**15.0**	**83.0**
Wav2vec2 Large	Llama-3.1-8B	**57.9**	11.8	19.6	**0.7**	6.3	11.9	81.8

Table 1: Performance comparisons of different cascaded ASR+LLM frameworks on mispronunciation detection and suggestion generation. Results show that, with the same LLM, using a small ASR model often leads to better performance. Overall, the cascaded ASR+LLM framework struggles with this task.

incorrect words, thus allowing LLMs to infer mispronunciation errors based on these transcription inconsistencies.

LLM-based In-Context Learning. To enable LLMs to detect mispronunciation and generate targeted feedback, we prompt LLMs to conduct in-context learning using the one-shot demonstration. Specifically, LLMs are provided with the canonical text $\boldsymbol{W}_{1:N}$ alongside the ASR-generated transcription $\hat{\boldsymbol{W}}_{1:\hat{N}}$, along with one example illustrating how to identify mispronunciations by comparing discrepancies between the two texts. The LLMs then generate pronunciation feedback for each detected mispronounced word. An illustration of the system prompt and one-shot demonstration is provided in Figure 5 in the Appendix.

4.2 Evaluation Results

To assess the performance of the ASR + LLMs cascade framework in mispronunciation detection and suggestion generation, we evaluate the instruct versions of Mistral-7B (Jiang et al., 2023) and Llama-3.1-8B (Dubey et al., 2024) as the LLMs. For ASR models, we evaluate various sizes of Whisper (Small, Medium, Large) (Radford et al., 2022) and Wav2vec2[1] (Base, Large) (Baevski et al., 2020). The evaluation results are reported in Table 1.

Stronger ASR models degrade detection performance with the same LLM. Surprisingly, we observe that Whisper Small outperforms Whisper Medium and Whisper Large in the F1 score, and Wav2vec2 Base surpasses Wav2vec2 Large when paired with either Mistral-7B or Llama-3.1-8B. We conjecture that stronger ASR models tend to correct pronunciation errors during transcription due to their robustness to accent variations, preventing them from accurately reflecting learners' speech errors. Additionally, Wav2vec2 Base achieves better performance than Whisper Small, likely due to the Whisper's decoder introducing linguistic biases during decoding, whereas Wav2vec2 relies solely on greedy search with an encoder-only structure.

Stronger LLMs improve detection and feedback generation. For a given ASR model, LLama-3.1-8B consistently outperforms Mistral-7B in both mispronunciation detection and suggestion generation, achieving up to a 121.5% relative improvement in F1 score. This suggests that more capable LLMs, with stronger instruction-following abilities and richer commonsense knowledge, generalize better when prompted for a new task. However, LLama-3.1-8B also displays higher extra word rates compared to Mistral-7B, indicating an increased propensity for hallucination.

Despite these improvements, the overall performance remains suboptimal, highlighting the inherent limitations of the ASR+LLM cascade framework. This section underscores the need for further exploration beyond the cascaded ASR+LLM framework. The results presented here serve as a baseline for comparative studies in the following sections.

Limitations: ASR models discard acoustic information in their text outputs, restricting LLMs from further understanding the input speech and performing more complex downstream speech-related tasks.

[1]We use the CTC versions of Wav2vec2 Base and Large fine-tuned for ASR task.

ALMs	Mispronunciation Detection				Suggestion Generation		
	Precision ↑	Recall ↑	F1 ↑	EWR ↓	BLEU-2 ↑	ROUGE-L ↑	BERTScore ↑
Qwen-Audio	50.4	18.7	27.2	0.7	3.9	11.8	82.7
Qwen2-Audio	41.7	22.0	28.8	2.0	6.9	18.3	82.9
GPT-4o-Audio∗	**52.7**	**41.3**	**46.3**	**0.2**	**10.9**	**22.3**	**86.0**

Table 2: Performance comparisons of existing ALMs on mispronunciation detection and suggestion generation. Here ∗ indicates the use of a one-shot multimodal demonstration (audio and text), which slightly improves performance. While open-source ALMs outperform cascaded ASR+LLM frameworks, they remain suboptimal compared to GPT-4o-Audio.

5 Investigating Existing ALMs

To mitigate the loss of acoustic information, such as phonetic details during transcription in the framework of ASR + LLM cascade, we explore how existing ALMs perform chat-base pronunciation training in an end-to-end manner in this section. Typically, ALMs integrate an audio encoder and an LLM, where the audio representation is projected into the text embedding space through joint learning on both modalities. The audio encoder preserves acoustic information in latent audio representations, enabling the LLM to better understand speech characteristics compared to ASR-transcribed text.

5.1 Employed ALMs

We evaluate five ALMs including four open-source models: Pengi (Deshmukh et al., 2023), SpeechGPT (Zhang et al., 2023), Qwen-Audio (Chu et al., 2023), Qwen2-Audio (Chu et al., 2024), and one proprietary model: GPT-4o-Audio (Hurst et al., 2024). Each model receives text prompts along with corresponding audio input and then generates text-based responses. Example prompts can be found in Figure 6 (Qwen-Audio & Qwen2-Audio) and Figure 7 (GPT4o-Audio) in the Appendix.

5.2 Evaluation Results

Failure of Pengi and SpeechGPT. Interestingly, only Qwen-Audio, Qwen2-Audio, and GPT-4o-Audio can successfully follow the given instructions and perform chat-based pronunciation training. In contrast, Pengi and SpeechGPT struggle with this task, either generating irrelevant responses or misinterpreting it as ASR, failing to detect mispronunciations and generate suggestions. Figure 8 in the Appendix illustrates failure cases from Pengi and SpeechGPT, highlighting the significance of strong instruction-following capability for complex downstream audio-language tasks.

ALMs outperform cascaded ASR+LLM on pronunciation training. Table 2 presents the evaluation results for Qwen-Audio, Qwen2-Audio, and GPT-4o-Audio. Notably, Qwen2-Audio, despite lacking task-specific fine-tuning, outperforms all cascaded ASR+LLM approaches, demonstrating the superiority of end-to-end ALMs with audio encoders that preserve acoustic information in latent representations. GPT-4o-Audio further improves performance, achieving 60.8% relative F1 improvement over Qwen2-Audio, showcasing its stronger capability and better generalization to unseen new audio-language tasks.

While GPT-4o-Audio achieved state-of-the-art results so far, its closed-source nature and potentially large model size present challenges. Bridging the performance gap between GPT-4o-Audio and open-source ALMs remains worth being further investigated.

> **Limitations:** Despite notable improvements, open-source ALMs still lag behind GPT-4o-Audio, as they are not explicitly trained for mispronunciation detection and suggestion generation.

6 Instruction Tuning ALMs for Interactive Language Learning

As discussed in Section 4 and Section 5, the cascaded ASR+LLM framework and existing ALMs exhibit notable limitations in performing chat-based pronunciation training, particularly in their inability to accurately detect mispronunciations and generate actionable suggestions. To address these challenges, this section focuses on enabling end-to-end ALMs to effectively perform this task. Specifically, we construct a synthesized training dataset and investigate its potential to enhance chat-based pronunciation training in ALMs. We build ALMs by leveraging well-trained audio encoders and LLMs while facilitating modality

Audio Encoders	LLMs	Mispronunciation Detection				Suggestion Generation		
		Precision ↑	Recall ↑	F1 ↑	EWR ↓	BLEU-2 ↑	ROUGE-L ↑	BERTScore ↑
ASR+LLM Cascade SOTA								
Wav2vec2 Base + Llama-3.1-8B		**53.8**	17.8	26.8	1.1	7.3	15.0	83.0
Existing ALM SOTA								
GPT-4o-Audio		52.7	41.3	46.3	0.2	10.9	22.3	86.0
Instruction-Tuned ALMs								
Whisper Small	Mistral-7B	50.5	65.5	57.1	0.0	17.4	25.9	85.7
Whisper Medium	Mistral-7B	51.6	78.2	62.1	0.0	19.7	30.7	87.2
Whisper Large	Mistral-7B	50.6	81.8	62.5	0.0	20.1	30.5	87.2
Whisper Small	Llama-3.1-8B	49.7	68.2	57.5	0.0	17.2	25.4	85.5
Whisper Medium	Llama-3.1-8B	51.2	78.3	61.9	0.0	**20.4**	**31.9**	**87.4**
Whisper Large	Llama-3.1-8B	48.9	**87.7**	**62.8**	0.0	20.0	30.5	87.3

Table 3: Performance comparisons of our instruction-tuned ALMs with the state-of-te-art baselines in Section 4 and Section 5 on mispronunciation detection and suggestion generation. It is noted that our instruction-tuned ALMs significantly outperform the baselines, even including GPT-4o-Audio. Besides, with the same LLM backbone, the ALM with a larger-sized audio encoder tends to perform better.

fusion through audio modality alignment and task-specific speech instruction tuning. An overview of the framework is illustrated in Figure 2.

6.1 Speech Instruction Tuning

Since LLMs inherently lack an understanding of the audio input, a trainable projector is introduced to align the acoustic features extracted from audio encoders with the text embedding space. This projector consists of two linear layers with a GeLU activation function (Hendrycks and Gimpel, 2016). Then we prepare data to instruction tune the resulted ALMs. Inspired by Liu et al. (2023), we conduct two-stage training, including a stage of acoustic feature alignment and a stage of task-specific instruction tuning.

Stage 1: Acoustic feature alignment. As the training data for chat-based pronunciation training are limited, we leverage the abundance of ASR data for the first stage. Specifically, we sample 200k pairs of audio and corresponding text transcription from the English subset of Common-Voice (Ardila et al., 2020). Then we prepare the instruction format as a prompt-response pair. The prompt includes a question related to ASR and the audio while the response is the text transcription for the audio. Examples of these constructed question-answer pairs are provided in Figure 9 in the Appendix. Then the training objective is the auto-regressive loss on the response part. We employ a learning rate of 1e-3, a batch size of 256, and a training duration of one epoch. It is noted that only the projector is trainable at this stage.

Stage 2: Task-specific instruction tuning. In this stage, we continue instruction tuning ALMs on the data of chat-based pronunciation training. Similar to the curation procedure of L2-Arctic-plus, we construct 2.7k prompt-response pairs based on the L2-Arctic dataset. The prompt includes the system prompt for the LLM backbone, a question to prompt LLMs to detect mispronunciations and provide actionable suggestions, and the audio. Then the ground-truth response is a sequence of word-level error-suggestion pairs generated by GPT-4o. It is noted that during the curation, we exclude the samples used to construct L2-Arctic-plus and there is no human verification in this process. The prompt example is presented in Figure 10. The training objective is still the auto-regressive loss on the response part. In this stage, we fine-tune both the projector and the LLM backbone. To mitigate the high computational burden associated with the full fine-tuning of large models, we adopt LoRA (Hu et al., 2021) tuning with a learning rate of 9e-4, a batch size of 128, and a training duration of 5 epochs. Our empirical analysis demonstrates that LoRA tuning is sufficient to highlight the value of the dataset and the potential benefits of task-specific speech instruction tuning.

6.2 Evaluation Results

We construct our instruction-tuned ALMs considering different LLM backbones and different Whisper encoders, following Section 4. Afterward, we compare the performance of these models with the best-performing baselines in Table 3.

Instruction-tuned ALMs outperform baseline methods. The empirical results in Table 3 indicate that our instruction-tuned ALMs surpass the state-of-the-art ASR+LLM cascade framework and existing ALM, achieving relative improvements of up to 134.3% and 35.6% in F1 score, respectively. Notably, our ALMs could even outperform GPT-4o-Audio. Furthermore, the performance on suggestion generation exhibits substantial enhancements after task-specific instruction tuning, as reflected in BLEU-2, ROUGE-L, and BERTScore metrics. These results further underscore the efficacy of task-specific instruction tuning and highlight the significance of the utilized dataset.

Instruction tuning mitigates hallucination in mispronunciation detection. Notably, the empirical results reveal a significant reduction in EWRs, indicating that extraneous words outside the canonical text do not appear in the detection outputs. This suggests that task-specific instruction tuning effectively mitigates hallucination in mispronunciation detection by reinforcing a focus on words within the canonical text.

Larger audio encoders yield improved detection performance. The results further demonstrate that employing large audio encoders leads to enhanced mispronunciation detection performances in terms of F1 score. This improvement is likely attributed to the increased embedding space in large audio encoders, which facilitates more effective fine-tuning. Additionally, a comparison of Mistral-7B and Llama-3.1-8B with the same audio encoder reveals comparable performance in both detection and generation, despite differences in the underlying LLMs. These findings contrast with those observed in the cascaded ASR+LLM framework, emphasizing the critical role of task-specific instruction tuning in enabling ALMs to handle more complex tasks.

6.3 Human Evaluation

Setups. To validate the previously observed results, we conduct a human evaluation. For this purpose, we randomly select 2 audio samples per speaker from the L2-Arctic-plus dataset, resulting in a total of 12 audio samples for evaluation. Details regarding these samples are provided in Table 6 in the Appendix. The evaluation compares the responses generated by four models used in our earlier experiments: (a) Wav2vec2 Base + Llama-3.1-8B (ASR+LLM cascade); (b) Qwen2-Audio (ALMs); (c) GPT-4o-Audio (ALMs); (d) Whisper Large + Llama-3.1-8B (our instruction tuned ALMs). Seven participants were recruited to rate the models' outputs on three dimensions: suggestion relevance (**SR**), user understandability (**UU**), and overall evaluation (**OE**), using integer scores ranging from 1 to 5 (very bad, bad, neutral, good, very good). For each dimension, the final score of a model was determined by averaging scores from all participants across all 12 samples.

Method	SR ↑	UU ↑	OE ↑
(a) Wav2vec2 Base+Llama-3.1-8B	1.80	2.50	1.90
(b) Qwen2-Audio	2.12	2.83	2.26
(c) GPT-4o-Audio	2.88	3.51	3.07
(d) Whisper Large+Llama-3.1-8B	**3.80**	**3.81**	**3.73**

Table 4: Performance comparisons of our instruction-tuned ALMs with the baselines in Section 4 and Section 5 by human evaluations. Here **SR** refers to suggestion relevance, **UU** refers to user understandability, and **OE** refers to overall evaluation.

Results. The evaluation results are summarized in Table 4. The findings indicate that our instruction-tuned ALM outperforms other models across all three evaluation dimensions. Notably, when compared to GPT-4o-Audio, our instruction-tuned model achieves substantial improvements of 24.2%, 8.5% and 21.5% in SR, UU, and OE, respectively. The superior performance of our instruction-tuned model in the SR metric suggests that the generated suggestions are clearer, more practical, and actionable. This clarity and relevance likely contribute to the higher overall evaluation score attributed to the generated content.

7 Conclusion

In this paper, we explore the untapped potential of ALMs in enhancing chat-based pronunciation training for second-language learners. By introducing the L2-Arctic-plus dataset, which includes detailed annotations for pronunciation errors along with actionable feedback, we benchmark cascaded ASR+LLM frameworks and existing ALMs on this task. Furthermore, we improve both mispronunciation detection and feedback generation by instruction-tuning ALMs on L2-Arctic-plus, which outperform state-of-the-art baselines. Our findings underscore the value of the proposed dataset and extend the application of ALMs in interactive chat-based pronunciation training, advancing them as more effective tools for education purposes.

Limitations

While our work demonstrates significant advancements in chat-based pronunciation training through instruction tuning ALMs on L2-Arctic-plus, several limitations remain that warrant further investigation and improvement. First, the current chat-based pronunciation training primarily targets "reading-aloud" pronunciation training scenarios. Future research could expand its scope to include free-form conversational scenarios, enabling a broader assessment of language use beyond pronunciation training to support more comprehensive language learning. Second, the feedback generated in this work is provided solely in text format, which, while informative, may lack the intuitiveness of auditory feedback. Future efforts could explore generating responses in other modalities, such as high-quality synthesized speech or golden speech as pronunciation references, to enhance learners' understanding and engagement during training. Addressing these limitations would further refine the capabilities of ALMs in interactive language learning applications.

References

Moustafa Amrate and Pi-hua Tsai. 2024. Computer-assisted pronunciation training: A systematic review. *ReCALL*, pages 1–21.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France. European Language Resources Association.

Vipul Arora, Aditi Lahiri, and Henning Reetz. 2018. Phonological feature-based speech recognition system for pronunciation training in non-native language learning. *The Journal of the Acoustical Society of America*, 143(1):98–108.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Preprint*, arXiv:2006.11477.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*.

Soham Deshmukh, Benjamin Elizalde, Rita Singh, and Huaming Wang. 2023. Pengi: An audio language model for audio tasks. *Advances in Neural Information Processing Systems*, 36:18090–18108.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, and Abhishek et al. Kadian. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Maxine Eskenazi. 2009. An overview of spoken language technology for education. *Speech Communication*, 51(10):832–844.

Yiqing Feng, Guanyu Fu, Qingcai Chen, and Kai Chen. 2020. Sed-mdd: Towards sentence dependent end-to-end mispronunciation detection and diagnosis. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3492–3496.

Xuelong Geng, Tianyi Xu, Kun Wei, Bingshen Mu, Hongfei Xue, He Wang, Yangze Li, Pengcheng Guo, Yuhang Dai, Longhao Li, Mingchen Shao, and Lei Xie. 2024. Unveiling the potential of llm-based asr on chinese open-source datasets. In *2024 IEEE 14th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pages 26–30.

Yuan Gong, Ziyi Chen, Iek-Heng Chu, Peng Chang, and James Glass. 2022. Transformer-based multi-aspect multi-granularity non-native english speaker pronunciation assessment. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7262–7266. IEEE.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Lynn E Henrichsen. 2021. An illustrated taxonomy of online capt resources. *RELC Journal*, 52(1):179–188.

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Wenping Hu, Yao Qian, Frank K Soong, and Yong Wang. 2015. Improved mispronunciation detection with deep neural network trained acoustic models and transfer learning based logistic regression classifiers. *Speech Communication*, 67:154–166.

Rongjie Huang, Mingze Li, Dongchao Yang, Jiatong Shi, Xuankai Chang, Zhenhui Ye, Yuning Wu, Zhiqing Hong, Jiawei Huang, Jinglin Liu, et al. 2024. Audiogpt: Understanding and generating speech, music, sound, and talking head. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 23802–23804.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Yassine El Kheir, Ahmed Ali, and Shammur Absar Chowdhury. 2023. Automatic pronunciation assessment–a review. *arXiv preprint arXiv:2310.13974*.

Daniel Korzekwa, Jaime Lorenzo-Trueba, Szymon Zaporowski, Shira Calamaro, Thomas Drugman, and Bozena Kostek. 2021. Mispronunciation detection in non-native (l2) english with uncertainty modeling. In *ICASSP 2021-2021 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 7738–7742. IEEE.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Ziyang Ma, Yifan Yang, Yifan Yang, Zhifu Gao, Jiaming Wang, Zhihao Du, Fan Yu, Qian Chen, Siqi Zheng, Shiliang Zhang, and Xie Chen. 2024. An embarrassingly simple approach for llm with strong asr capacity. *ArXiv*, abs/2402.08846.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Linkai Peng, Kaiqi Fu, Binghuai Lin, Dengfeng Ke, and Jinsong Zhang. 2021. A study on fine-tuning wav2vec2. 0 model for the task of mispronunciation detection and diagnosis. In *Interspeech*, pages 4448–4452.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. *Preprint*, arXiv:2212.04356.

Pamela M Rogerson-Revell. 2021. Computer-assisted pronunciation training (capt): Current issues and future directions. *RELC Journal*, 52(1):189–205.

Silke M Witt and Steve J Young. 2000. Phone-level pronunciation scoring and assessment for interactive language learning. *Speech communication*, 30(2-3):95–108.

Xiaoshuo Xu, Yueteng Kang, Songjun Cao, Binghuai Lin, and Long Ma. 2021. Explore wav2vec 2.0 for mispronunciation detection. In *Interspeech*, pages 4428–4432.

Mu Yang, Kevin Hirschi, Stephen D Looney, Okim Kang, and John HL Hansen. 2022. Improving mispronunciation detection with wav2vec2-based momentum pseudo-labeling for accentedness and intelligibility assessment. *arXiv preprint arXiv:2203.15937*.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*.

Junbo Zhang, Zhiwen Zhang, Yongqing Wang, Zhiyong Yan, Qiong Song, Yukai Huang, Ke Li, Daniel Povey, and Yujun Wang. 2021. speechocean762: An open-source non-native english speech corpus for pronunciation assessment. *arXiv preprint arXiv:2104.01378*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Guanlong Zhao, Evgeny Chukharev-Hudilainen, Sinem Sonsaat, Alif Silpachai, Ivana Lucic, Ricardo Gutierrez-Osuna, and John Levis. 2018. L2-arctic: A non-native english speech corpus.

A Experimental Details

A.1 Implementations Details

Our implementation leverages PyTorch and HuggingFace. The models used in the experiments, along with their associated versions and resources, are summarized in Table 5. The experiments are conducted using 2× NVIDIA RTX A40 GPUs. For decoding, we set the maximum new tokens to 1024, the temperature to 0.6, and the top_p to 0.9.

Evaluating each model on the entire L2-Arctic-plus dataset typically requires 4–6 GPU hours. For instruction tuning of our ALMs, the acoustic feature alignment stage takes approximately 12–14 GPU hours, whereas the task-specific instruction tuning stage requires around 4–6 GPU hours.

A.2 Prompting and output parsing designs

The prompt templates for cascaded ASR+LLM frameworks, Qwen-Audio & Qwen2-Audio, and our instruction-tuned ALMs are shown in Figure 5, Figure 6, and Figure 10, respectively.

For the cascaded ASR+LLM frameworks, all words from the input are outputted. If a word is not identified as mispronounced, both the issue and suggestion fields are marked as "None". This design ensures consistency with our output format. For the rest methods, only the words detected as mispronounced, along with their corresponding issues and suggestions, are included in the output.

Given these two different output formats, we implement two corresponding parsing strategies, with further subtle adjustments for each specific model tendency. For example, Qwen2-Audio often appends "No Problem" to its output, which is removed during processing. Additionally, both Qwen2-Audio and Qwen-Audio may include words marked as correct but accompanied by "No issues" in the issue and suggestion fields. Such words are excluded from the analysis.

To handle duplicate output, we retain only unique entries, ensuring consistency in evaluation. Models sometimes fail to strictly adhere strictly to the specified format, introducing unnecessary explanations before or after their responses. To address this, we apply pattern-matching techniques based on the defined format to extract only the relevant portions.

After applying these processing steps, the final parsed output, as illustrated in Figure 11, is generated. This parsed output is used as the standardized input for evaluation across all models.

B Failure Cases of Pengi and SpeechGPT

To assess the performance of existing ALMs on this task, we test Pengi, SpeechGPT, Qwen-Audio, Qwen2-Audio, and GPT-4o-Audio. Notably, Pengi and SpeechGPT fail to complete the task. To further analyze their limitations, we design two types of prompts. The *concise prompt* is a zero-shot simple instruction with no constraints on the output format, aiming at evaluating the model's basic task comprehension. The *full prompt* is similar to those used for Qwen-Audio, Qwen2-Audio, and GPT-4o-Audio, providing a one-shot instruction with strict output format requirements.

Both Pengi and SpeechGPT require specific modifications to their input format. For example, Pengi requires the addition of "question:" at the beginning of the prompt, while SpeechGPT necessitates appending the audio path in the format: "This is input: {audio_path}". Despite these adjustments, neither model successfully completes the task. Pengi generates meaningless text, and SpeechGPT defaulted to performing only automatic speech recognition (ASR), transcribing the audio input without regard to the task-specific prompt. Examples of the prompts and failure cases are presented in Figure 8.

C Human Evaluation

The audio paths and corresponding canonical texts selected from the L2-Arctic-plus dataset for human evaluation are listed in Table 6.

Model	Resource
Whisper-Small	https://huggingface.co/openai/whisper-small
Whisper-Medium	https://huggingface.co/openai/whisper-medium
Whisper-Large	https://huggingface.co/openai/whisper-large
Wav2vec2-Base	https://huggingface.co/facebook/wav2vec2-base-960h
Wav2vec2-Large	https://huggingface.co/facebook/wav2vec2-large-960h-lv60-self
Mistral-7B	https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1
Llama-3.1-8B	https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct
SpeechGPT	https://github.com/0nutation/SpeechGPT/tree/main/speechgpt
Pengi	https://github.com/microsoft/Pengi
Qwen-Audio	https://huggingface.co/Qwen/Qwen-Audio-Chat
Qwen2-Audio	https://huggingface.co/Qwen/Qwen2-Audio-7B-Instruct
GPT-4o-Audio	API gpt-4o-audio-preview Version
GPT-4o	API gpt-4o Version

Table 5: The overview of models used in this work.

Audio Path	Ground Truth Text / Canonical Text
NJS/wav/arctic_a0137.wav	Then he stepped back with a low cry of pleasure.
NJS/wav/arctic_b0279.wav	He gave one last snarl and slid from view among the trees.
TLV/wav/arctic_a0122.wav	Two years ago I gave up civilization for this.
TLV/wav/arctic_a0063.wav	Yes, it was a man who asked a stranger.
TNI/wav/arctic_a0282.wav	If you mean to insinuate, Brentwood began hotly.
TNI/wav/arctic_a0107.wav	If you only could know how I thank you.
TXHC/wav/arctic_a0075.wav	There has been a change, she interrupted him.
TXHC/wav/arctic_a0052.wav	It was a curious coincidence.
YKWK/wav/arctic_a0022.wav	Hardly were our plans made public before we were met by powerful opposition.
YKWK/wav/arctic_a0369.wav	In partnership with daylight, the pair raided the San Jose interurban.
ZHAA/wav/arctic_a0076.wav	The gray eyes faltered, the flush deepened.
ZHAA/wav/arctic_a0062.wav	The men stared into each other's face.

Table 6: Audio samples used for human evaluation from the L2-Arctic-plus dataset.

Ground Truth Generation Prompt (GPT-4o):

System Prompt:

```
You are a phonetics expert. I will provide text and annotations of a spoken utterance. Your task is to identify any
    ↪ pronunciation errors and suggest improvements. Use the following format for each word that contains a
    ↪ pronunciation error:

word [(Issue: Explanation) (Suggestion: How to improve using ARPAbet symbols)] [(Issue: Explanation) (Suggestion: How to
    ↪ improve using ARPAbet symbols)] [(Issue: Explanation) (Suggestion: How to improve using ARPAbet symbols)]...

Below is the phonetic annotation for the utterance. Each word includes the phonemes it contains and may have errors
    ↪ annotated as:
- Correct pronunciation: No changes in the forced-alignment labels.
- Substitution error: Format is 'CPL,PPL,s' (Correct Phoneme Label, Perceived Phoneme Label, Substitution). If it is
    ↪ hard to judge, 'err' is used. If there is a foreign accent, mark the perceived phoneme with a '*'.
- Addition error: Format is 'sil,PPL,a' (Silence, Perceived Phoneme Label, Addition).
- Deletion error: Format is 'CPL,sil,d' (Correct Phoneme Label, Silence, Deletion).

Important: You must strictly follow the annotations provided in the "annotation_info" field. Only report the errors
    ↪ explicitly indicated in the annotations. Do not add or remove errors based on assumptions or external knowledge.

Output Format:
- Only plain text without any Markdown, JSON, or code formatting symbols.
- Avoid extra newlines or spaces.
- If there are no errors, respond with exactly: No error (without quotes or additional characters).

Example input:
{
    "text": "But there came no promise from the bow of the canoe",
    "annotation_info": {
        "but": ["B", "AH", "T"],
        "there": ["DH, err, s", "EH", "R"],
        "came": ["K", "EY", "M"],
        "no": ["N", "OW"],
        "promise": ["P", "R", "AA", "M", "AH", "S"],
        "from": ["F", "R", "AH, AO, s", "M, N, s"],
        "the": ["DH, D, s", "AH, EH, s"],
        "bow": ["B", "OW, AW, s"],
        "of": ["sil, err, a", "AH, AO, s", "V, F, s"],
        "canoe": ["K", "AH", "N", "UW", "sil, IY, a"]
    }
}

Example output:
there [(Issue: "DH" was substituted with an unclear phoneme, indicating a substitution error) (Suggestion: Practice
    ↪ producing /DH/ by contrasting it with /D/ using ARPAbet words like "THE" (/DH AH/) vs. "DO" (/D UW/))]
from [(Issue: "AH" was replaced with "AO", indicating a vowel substitution) (Suggestion: Practice /AH/ vs. /AO/
    ↪ distinction with pairs like "CUT" (/K AH T/) vs. "CAUGHT" (/K AO T/))] [(Issue: "M" was replaced with "N",
    ↪ indicating a consonant substitution) (Suggestion: Practice bilabial nasal /M/ versus alveolar nasal /N/ using
    ↪ "SUM" (/S AH M/) vs. "SUN" (/S AH N/))]
the [(Issue: "DH" was replaced with "D", indicating a substitution error) (Suggestion: Strengthen the articulation of
    ↪ /DH/ by comparing it with /D/ in words like "THIS" (/DH IH S/) vs. "DIS" (/D IH S/))]
bow [(Issue: "OW" was replaced with "AW", indicating a substitution error) (Suggestion: Practice diphthongs /OW/ and
    ↪ /AW/ using pairs like "BOW" (/B OW/) vs. "BOUGH" (/B AW/))]
of [(Issue: An extra phoneme was added, suggesting an insertion error) (Suggestion: Focus on avoiding unnecessary vowel
    ↪ insertions by practicing smooth transitions between words)] [(Issue: "AH" was replaced with "AO", indicating a
    ↪ vowel substitution) (Suggestion: Practice /AH/ and /AO/ distinction using "HOT" (/HH AA T/) vs. "HAWED" (/HH AO
    ↪ D/))] [(Issue: "V" was replaced with "F", indicating a consonant substitution) (Suggestion: Practice voiced /V/
    ↪ versus voiceless /F/ using "VAN" (/V AE N/) vs. "FAN" (/F AE N/))]
canoe [(Issue: An extra "IY" was added, suggesting an insertion error) (Suggestion: Practice avoiding vowel insertion
    ↪ using controlled phrases, focusing on words like "CANOE" (/K AH N UW/))]
```

User Prompt:

```
Here is the phonetic annotation for an utterance:
"text": "{text}"
"annotation_info": {annotation_info}

Please identify the pronunciation errors and suggest improvements in the specified format: word1 [(Issue: Explanation)
    ↪ (Suggestion: How to improve using ARPAbet symbols)] [(Issue: Explanation) (Suggestion: How to improve using
    ↪ ARPAbet symbols)] [(Issue: Explanation) (Suggestion: How to improve using ARPAbet symbols)]...
word2[(Issue: Explanation) (Suggestion: How to improve using ARPAbet symbols)] [(Issue: Explanation) (Suggestion: How to
    ↪ improve using ARPAbet symbols)] [(Issue: Explanation) (Suggestion: How to improve using ARPAbet symbols)]...
...
For each word in "annotation_info", ensure that the number of [(Issue)(Suggestion)] pairs exactly matches the number of
    ↪ errors indicated for that word in "annotation_info". There must be no extra or missing pairs.
If there are no pronunciation errors, output "No error" without any extra words.
You must strictly follow the errors explicitly provided in the "annotation_info" field. Do not add or remove errors
    ↪ based on assumptions or external knowledge.
```

Figure 3: Ground truth generation prompt (GPT-4o).

Ground Truth Generation Example:

Input:

```
"text": "You're joking me sir the other managed to articulate",
"annotation_info": {
    "you're": ["Y", "UH", "R"],
    "joking": ["JH, ZH, s", "OW", "K", "IH", "NG", "sil, G, a", "sil, AH, a"],
    "me": ["M", "IY"],
    "sir": ["S", "ER, err, s"],
    "the": ["DH", "AH"],
    "other": ["AH", "DH, Z, s", "ER, err, s"],
    "managed": ["M", "AE", "N", "IH", "JH, ZH, s", "D"],
    "to": ["T", "UW"],
    "articulate": ["AA", "R, R*, s", "T", "IH", "K", "Y", "AH", "L", "EY, EH, s", "T"]
}
```

Original Output:

```
joking [(Issue: \"JH\" was replaced with \"ZH\", indicating a substitution error) (Suggestion: Practice the difference
    ↪ between /JH/ as in \"JOKE\" (/JH OW K/) and /ZH/ as in \"MEASURE\" (/M EH ZH ER/))] [(Issue: An extra \"G\"
    ↪ sound was added, indicating an addition error) (Suggestion: Focus on stopping after the /NG/ as in \"KING\" (/K
    ↪ IH NG/) without additional sounds)] [(Issue: An extra \"AH\" sound was added, indicating an addition error)
    ↪ (Suggestion: Avoid adding extra vowel sounds after completing the word)]\nsir [(Issue: Unclear pronunciation,
    ↪ \"ER\" perceived with uncertainty) (Suggestion: Practice /ER/ as in \"SIR\" (/S ER/) to add clarity)]\nother
    ↪ [(Issue: \"DH\" was replaced with \"Z\", indicating a substitution error) (Suggestion: Practice unvoiced /DH/ as
    ↪ in \"THIS\" (/DH IH S/) instead of voiced consonant sounds like /Z/)] [(Issue: Unclear pronunciation, \"ER\"
    ↪ perceived with uncertainty) (Suggestion: Practice /ER/ as in \"HER\" (/HH ER/) for more distinct
    ↪ articulation)]\nmanaged [(Issue: \"JH\" was replaced with \"ZH\", indicating a substitution error) (Suggestion:
    ↪ Practice the distinction between /JH/ as in \"JUDGE\" (/JH AH JH/) and /ZH/ as in \"VISION\" (/V IH ZH UH
    ↪ N/))]\narticulate [(Issue: \"R\" was replaced with a foreign-accented \"R*\", indicating a substitution error)
    ↪ (Suggestion: Practice the American /R/ sound as in \"RED\" (/R EH D/) emphasizing the retroflex position of the
    ↪ tongue)] [(Issue: \"EY\" was replaced with \"EH\", indicating a substitution error) (Suggestion: Practice the
    ↪ distinction between /EY/ as in \"DATE\" (/D EY T/) and /EH/ as in \"BET\" (/B EH T/))]
```

Parsed Output:

```
joking:
    issue: "JH" was replaced with "ZH", indicating a substitution error. An extra "G" sound was added, indicating an
        ↪ addition error. An extra "AH" sound was added, indicating an addition error.
    suggestion: Practice the difference between /JH/ as in "JOKE" (/JH OW K/) and /ZH/ as in "MEASURE" (/M EH ZH ER/).
        ↪ Focus on stopping after the /NG/ as in "KING" (/K IH NG/) without additional sounds. Avoid adding extra
        ↪ vowel sounds after completing the word.

sir:
    issue: Unclear pronunciation, \"ER\" perceived with uncertainty.
    suggestion: "Practice /ER/ as in \"SIR\" (/S ER/) to add clarity.

other:
    issue: "DH" was replaced with "Z", indicating a substitution error. Unclear pronunciation, "ER" perceived with
        ↪ uncertainty.
    suggestion: Practice unvoiced /DH/ as in "THIS" (/DH IH S/) instead of voiced consonant sounds like /Z/. Practice
        ↪ /ER/ as in "HER" (/HH ER/) for more distinct articulation.

managed:
    issue: "JH" was replaced with "ZH", indicating a substitution error.
    suggestion: Practice the distinction between /JH/ as in "JUDGE" (/JH AH JH/) and /ZH/ as in "VISION" (/V IH ZH UH N/)

articulate:
    issue: "R" was replaced with a foreign-accented "R*", indicating a substitution error. "EY" was replaced with "EH",
        ↪ indicating a substitution error.
    suggestion: Practice the American /R/ sound as in "RED" (/R EH D/) emphasizing the retroflex position of the tongue.
        ↪ Practice the distinction between /EY/ as in "DATE" (/D EY T/) and /EH/ as in \"BET\" (/B EH T/).
```

Figure 4: Ground truth generation example (GPT-4o).

Cascaded ASR+LLMs Prompt:

System Prompt:

```
You are a phonetics expert tasked with identifying pronunciation differences between the provided Ground Truth
and the corresponding pronunciation. Analyze each word in the Ground Truth, identify pronunciation issues,
and offer suggestions for improvement.
```

User Prompt:

```
You are a phonetics expert. Your task is to compare the provided Transcribed Text with the Ground Truth transcription.
Identify any pronunciation differences for each word in the Ground Truth based on the transcription and provide specific
suggestions for improvement.

Input:
Ground Truth: <ground_truth>
Transcribed Text: <transcribed_text>

Output Format:
word: <word_in_ground_truth>
issue: <issues>
suggestion: <suggestions>
...

Output Rules:
1. Analyze each word in the Ground Truth and compare it with the corresponding word in the Transcribed Text.
2. For each word in the Ground Truth, output:
   word: <word_in_ground_truth>
   issue: <issues> (if there are pronunciation issues)
   suggestion: <suggestions> (if there are pronunciation issues)
   If there are no issues with a word, output:
   word: <word_in_ground_truth>
   issue: None
   suggestion: None
3. If a word has multiple issues, combine them into a single issue line and provide a single combined suggestion
   for correction.
4. Do not include any additional commentary outside of the analysis and suggestions.
5. Use ARPAbet phonetic symbols to describe the pronunciation issues.

Example Input:
Ground Truth: you're joking me sir the other managed to articulate
Transcribed Text: your soking me ser the other managed to articulate

Example Output:
word: you're
issue: None
suggestion: None
...
word: articulate
issue: "R" was replaced with a foreign-accented "R*", indicating a substitution error. "EY" was replaced with "EH",
     ↪ indicating a substitution error.
suggestion: Practice the American /R/ sound as in "RED" (/R EH D/) emphasizing the retroflex position of the tongue.
     ↪ Practice the distinction between /EY/ as in "DATE" (/D EY T/) and /EH/ as in "BET" (/B EH T/)
```

Figure 5: Cascaded ASR+LLMs Prompt

Qwen-Audio and Qwen2-Audio Prompt:

System Prompt:

```
You are a phonetics expert tasked with analyzing the pronunciation of audio and comparing it to the provided Ground
    ↪ Truth text.
Your goal is to identify pronunciation issues, such as substitution, addition, or deletion of sounds, based on the audio
    ↪ input.

Instructions:
1. For each word in the Ground Truth, compare its pronunciation in the audio.
2. Identify any mispronunciations and describe the issue (substitution, addition, deletion of sounds).
3. For each issue, provide a suggestion using ARPAbet phonetic symbols.
4. If the pronunciation is correct, simply output "No Problem".
5. Do not include additional commentary. Just output the issues and suggestions for each word that has problems.

Your task is to analyze the following audio and Ground Truth text for pronunciation issues and provide your suggestions.
```

User Prompt:

```
You are a phonetics expert. Your task is to detect mispronouciation based on given Ground Truth and Audio.
This is an example of the format you should use and some output rules you should follow.

Output Format:
word: <word_in_ground_truth> issue: <issues> suggestion: <suggestions>
word: <word_in_ground_truth> issue: <issues> suggestion: <suggestions>
...

Output Rules:
1. Analyze each word in the Ground Truth and compare it with the pronunciation in the actual audio.
2. If the word in the Ground Truth has one or more pronunciation issues based on the audio:
    a. List the word from the Ground Truth.
    b. Combine all issues into a single line under "issue".
    c. Provide a single combined suggestion for correcting the issues using ARPAbet phonetic symbols.
3. Ensure the analysis focuses on the pronunciation of Ground Truth words as they appear in the audio.
4. Do not include any additional commentary outside of the analysis and suggestions. Just begin with the first
    ↪ mispronunced word, instead of using 'Output:'.
5. Use ARPAbet symbols and English to describe phonetic issues.
6. If there are no issues with the words in the Ground Truth, output 'No Problem'. "No Problem" should appear on its own
    ↪ and cannot be included as part of the issue or suggestion.

Here is an example of how you should analyze pronunciation based on the audio and the Ground Truth text.

Input:
Ground Truth: "you're joking me sir the other managed to articulate"
Output:
word: joking issue: "JH" was replaced with "ZH", indicating a substitution error. An extra "G" sound was added,
    ↪ indicating an addition error. An extra "AH" sound was added, indicating an addition error. suggestion: Practice
    ↪ the difference between /JH/ as in "JOKE" (/JH OW K/) and /ZH/ as in "MEASURE" (/M EH ZH ER/). Focus on stopping
    ↪ after the /NG/ as in "KING" (/K IH NG/) without additional sounds. Avoid adding extra vowel sounds after
    ↪ completing the word.
word: sir issue: Unclear pronunciation, "ER" perceived with uncertainty suggestion: Practice /ER/ as in "SIR" (/S ER/)
    ↪ to add clarity
word: other issue: "DH" was replaced with "Z", indicating a substitution error. Unclear pronunciation, "ER" perceived
    ↪ with uncertainty. suggestion: Practice unvoiced /DH/ as in "THIS" (/DH IH S/) instead of voiced consonant sounds
    ↪ like /Z/. Practice /ER/ as in "HER" (/HH ER/) for more distinct articulation.
word: managed issue: "JH" was replaced with "ZH", indicating a substitution error suggestion: Practice the distinction
    ↪ between /JH/ as in "JUDGE" (/JH AH JH/) and /ZH/ as in "VISION" (/V IH ZH UH N/)
word: articulate issue: "R" was replaced with a foreign-accented "R*", indicating a substitution error. "EY" was
    ↪ replaced with "EH", indicating a substitution error. suggestion: Practice the American /R/ sound as in "RED" (/R
    ↪ EH D/) emphasizing the retroflex position of the tongue. Practice the distinction between /EY/ as in "DATE" (/D
    ↪ EY T/) and /EH/ as in "BET" (/B EH T/)

Input:
Ground Truth: {ground_truth}
Audio: {audio_input}
Output:
```

Figure 6: Qwen-Audio and Qwen2-Audio Prompt

GPT4o-Audio Prompt:

System Prompt:

```
You are a phonetics expert tasked with identifying pronunciation differences between the provided Ground Truth and the
    ↪ corresponding pronunciation.
Analyze each word in the Ground Truth, identify pronunciation issues, and offer suggestions for improvement.
```

User Prompt:

```
You are a phonetics expert. Your task is to detect mispronouciation based on given Ground Truth and Audio.
This is an example of the format you should use and some output rules you should follow.

Output Format:
word: <one_word_in_ground_truth> issue: <issues> suggestion: <suggestions>
word: <one_word_in_ground_truth> issue: <issues> suggestion: <suggestions>
...

Output Rules:
1. Analyze each word in the Ground Truth and compare it with the pronunciation in the actual audio.
2. If the word in the Ground Truth has one or more pronunciation issues based on the audio:
    a. List the word from the Ground Truth.
    b. Combine all issues into a single line under "issue".
    c. Provide a single combined suggestion for correcting the issues using ARPAbet phonetic symbols.
3. If no errors are found in any of the Ground Truth words, output "No Problem". But there is a high probability of
    ↪ pronunciation problems.
4. Do not output anything except for the words with pronunciation issues or "No Problem".
5. Ensure the analysis focuses on the pronunciation of Ground Truth words as they appear in the audio.
6. Do not include any additional commentary outside of the analysis and suggestions.
7. Use ARPAbet symbols to describe phonetic issues.

Here is an example of how you should analyze pronunciation based on the audio and the Ground Truth text.

Input:
Ground Truth: "you're joking me sir the other managed to articulate"
Audio: <example_audio_input>
Output:
word: joking issue: "JH" was replaced with "ZH", indicating a substitution error. An extra "G" sound was added,
    ↪ indicating an addition error. An extra "AH" sound was added, indicating an addition error. suggestion: Practice
    ↪ the difference between /JH/ as in "JOKE" (/JH OW K/) and /ZH/ as in "MEASURE" (/M EH ZH ER/). Focus on stopping
    ↪ after the /NG/ as in "KING" (/K IH NG/) without additional sounds. Avoid adding extra vowel sounds after
    ↪ completing the word.
word: sir issue: Unclear pronunciation, "ER" perceived with uncertainty suggestion: Practice /ER/ as in "SIR" (/S ER/)
    ↪ to add clarity
word: other issue: "DH" was replaced with "Z", indicating a substitution error. Unclear pronunciation, "ER" perceived
    ↪ with uncertainty. suggestion: Practice unvoiced /DH/ as in "THIS" (/DH IH S/) instead of voiced consonant sounds
    ↪ like /Z/. Practice /ER/ as in "HER" (/HH ER/) for more distinct articulation.
word: managed issue: "JH" was replaced with "ZH", indicating a substitution error suggestion: Practice the distinction
    ↪ between /JH/ as in "JUDGE" (/JH AH JH/) and /ZH/ as in "VISION" (/V IH ZH UH N/)
word: articulate issue: "R" was replaced with a foreign-accented "R*", indicating a substitution error. "EY" was
    ↪ replaced with "EH", indicating a substitution error. suggestion: Practice the American /R/ sound as in "RED" (/R
    ↪ EH D/) emphasizing the retroflex position of the tongue. Practice the distinction between /EY/ as in "DATE" (/D
    ↪ EY T/) and /EH/ as in "BET" (/B EH T/)

Input:
Ground Truth: {ground_truth}
Audio: <audio_input>
Output:
```

Figure 7: GPT4o-Audio Prompt

Prompts and Failure Cases:

Prompt (concise version):

```
The ground truth of the audio is "Hardly were our plans made public before we were met by powerful opposition". Identify
    ↪ any mispronounced words, categorize the type of mispronunciation (substitude, addtion, delation), and provide
    ↪ suggested corrections.
```

Prompt (full version):

```
word: sir issue: Unclear pronunciation, "ER" perceived with uncertainty suggestion: Practice /ER/ as in "SIR" (/S ER/)
    ↪ to add clarity
word: other issue: "DH" was replaced with "Z", indicating a substitution error. Unclear pronunciation, "ER" perceived
    ↪ with uncertainty. suggestion: Practice unvoiced /DH/ as in "THIS" (/DH IH S/) instead of voiced consonant sounds
    ↪ like /Z/. Practice /ER/ as in "HER" (/HH ER/) for more distinct articulation.
word: managed issue: "JH" was replaced with "ZH", indicating a substitution error suggestion: Practice the distinction
    ↪ between /JH/ as in "JUDGE" (/JH AH JH/) and /ZH/ as in "VISION" (/V IH ZH UH N/)
word: articulate issue: "R" was replaced with a foreign-accented "R*", indicating a substitution error. "EY" was
    ↪ replaced with "EH", indicating a substitution error. suggestion: Practice the American /R/ sound as in "RED" (/R
    ↪ EH D/) emphasizing the retroflex position of the toYou are a phonetics expert. Your goal is to identify
    ↪ pronunciation issues, such as substitution, addition, or deletion of sounds, based on the audio input and Audio.
This is an example of the format you should use and some output rules you should follow.

Output Format:
word: <word_in_ground_truth> issue: <issues> suggestion: <suggestions>
word: <word_in_ground_truth> issue: <issues> suggestion: <suggestions>
...

Output Rules:
1. Analyze each word in the Ground Truth and compare it with the pronunciation in the actual audio.
2. If the word in the Ground Truth has one or more pronunciation issues based on the audio:
    a. List the word from the Ground Truth.
    b. Combine all issues into a single line under "issue".
    c. Provide a single combined suggestion for correcting the issues using ARPAbet phonetic symbols.
3. Ensure the analysis focuses on the pronunciation of Ground Truth words as they appear in the audio.
4. Do not include any additional commentary outside of the analysis and suggestions. Just begin with the first
    ↪ mispronunced word, instead of using 'Output:'.
5. Use ARPAbet symbols and English to describe phonetic issues.
6. If there are no issues with the words in the Ground Truth, output 'No Problem'. "No Problem" should appear on its own
    ↪ and cannot be included as part of the issue or suggestion.

Here is an example of how you should analyze pronunciation based on the audio and the Ground Truth text.

Input:
Ground Truth: "you're joking me sir the other managed to articulate"

Output:
word: joking issue: "JH" was replaced with "ZH", indicating a substitution error. An extra "G" sound was added,
    ↪ indicating an addition error. An extra "AH" sound was added, indicating an addition error. suggestion: Practice
    ↪ the difference between /JH/ as in "JOKE" (/JH OW K/) and /ZH/ as in "MEASURE" (/M EH ZH ER/). Focus on stopping
    ↪ after the /NG/ as in "KING" (/K IH NG/) without additional sounds. Avoid adding extra vowel sounds after
    ↪ completing the word.
ngue. Practice the distinction between /EY/ as in "DATE" (/D EY T/) and /EH/ as in "BET" (/B EH T/)

Input:
Ground Truth: "Hardly were our plans made public before we were met by powerful opposition"

Output:
```

Failure Case:

```
Pengi: (should add question: before prompt)
Input: question: + Prompt (concise version)
Output: mm
Input: question: + Prompt (full version)
Output: male

SpeechGPT: (should add the path of the audio input file at the end of prompt)
Input: Prompt (concise version) + "This is input: /NJS/wav/arctic_a0022.wav"
Output: ird's work our plans made public before we were met by powerful opposition
Input: Prompt (full version) + "This is input: /NJS/wav/arctic_a0022.wav"
Output: ighly were our plans made public before we were met by powerful opposition
```

Figure 8: Prompts and Failure Cases of Pengi and SpeechGPT

Question-Answer Pairs for Audio Modality Alignment:

Questions (User Prompt):

```
Q1:  Repeat the content of the audio <speech>
Q2:  Transcribe <speech>
Q3:  What is being said in <speech>
Q4:  Can you interpret <speech>?
Q5:  Please convert <speech> into text
Q6:  What does <speech> say?
Q7:  Could you transcribe <speech> for me?
Q8:  I need the text of <speech>
Q9:  Can you write out <speech>?
Q10: What's the content of <speech>?
Q11: Please provide the transcript of <speech>
Q12: Can you decode <speech>?
Q13: What is the transcription of <speech>?
Q14: Can you jot down <speech>?
Q15: What is the written form of <speech>?
Q16: Can you scribe <speech>?
```

Question-Answer Pairs examples:

```
user(Q): Can you decode <speech>?
assistant(A): Later he appeared in over forty films, playing a wide variety of characters.

user(Q): What is the written form of <speech>?
assistant(A): The only living species is the walrus.

user(Q): Can you decode <speech>?
assistant(A): This income level is higher than the county, state, and national median income levels.
```

Figure 9: Question-Answer Pairs for Audio Modality Alignment

Our Method Prompt:

System Prompt:

```
You are a phonetics expert tasked with identifying pronunciation differences between the provided Ground Truth and the
    ↪ corresponding pronunciation.
Analyze each word in the Ground Truth, identify pronunciation issues, and offer suggestions for improvement.
```

User Prompt:

```
Your task is to analyze the provided audio and compare it with the Ground Truth to identify pronunciation differences at
    ↪ the phoneme level.
The audio in <speech> contains a recording by a non-native English speaker.

Below is the ground truth transcription:
Ground Truth: "{ground_truth}"

Output Format:
word: <mispronounced_word> issue: <issues> suggestion: <suggestions>
word: <mispronounced_word> issue: <issues> suggestion: <suggestions>
word: <mispronounced_word> issue: <issues> suggestion: <suggestions>
...
word: <mispronounced_word> issue: <issues> suggestion: <suggestions>

Output Rules:
1. Analyze each word in the Ground Truth and compare it with the pronunciation in the actual audio.
2. If the word in the Ground Truth has one or more pronunciation issues based on the audio:
    a. List the word from the Ground Truth.
    b. Combine all issues into a single line under "issue".
    c. Provide a single combined suggestion for correcting the issues using ARPAbet phonetic symbols.
3. If no errors are found in any of the Ground Truth words, output "No Problem".
4. Do not output anything except for the words with pronunciation issues or "No Problem".
5. Ensure the analysis focuses on the pronunciation of Ground Truth words as they appear in the audio.
6. Do not include any additional commentary outside of the analysis and suggestions.
7. Use ARPAbet symbols to describe phonetic issues.
```

Figure 10: Our Method Prompt

Inference Output:

(a) Wav2Vec2 Base + Llama-3.1-8B:

```
stared:
    Issue: "ST" was replaced with "STIRD", indicating a substitution error. "D" was replaced with "D", indicating a
        ↪ substitution error. However, it seems like the word was pronounced as "stirred" instead of "stared", which
        ↪ is a different word.
    Suggestion: Practice the difference between the words "stared" (/ST EY R D/) and "stirred" (/ST ER D/).
other's:
    Issue: Unclear pronunciation, "ER" perceived with uncertainty
    Suggestion: Practice /ER/ as in "HER" (/HH ER/) for more distinct articulation.
```

(b) Qwen2-Audio (ALMs):

```
stared:
    Issue: "TH" was replaced with "D", indicating a substitution error. "ER" was replaced with "AHR", indicating a
        ↪ substitution error.
    Suggestion: Practice the distinction between /TH/ as in "THAT" (/TH AHT/) and /D/ as in "DAY". Also, practice the
        ↪ distinction between /ER/ as in "FAR" (/F ER/) and /AHR/ as in "HURR".
```

(c) GPT4o-Audio (ALMs):

```
men:
    Issue: "EH" was replaced with "IH", indicating a substitution error.
    Suggestion: Practice the vowel sound /EH/ as in "MEN" (/M EH N/), differentiating it from /IH/ as in "MIN" (/M IH
        ↪ N/).
stared:
    Issue: "EH" was replaced with "EY", indicating a substitution error.
    Suggestion: Focus on practicing /EH/ as in "BED" (/B EH D/), being careful not to elevate the tongue to make a /EY/
        ↪ sound as in "BADE" (/B EY D/).
into:
    Issue: "IH" was replaced with "EE", indicating a substitution error.
    Suggestion: Practice the /IH/ vowel as in "HIT" (/HH IH T/), avoiding the longer /EE/ vowel sound as in "HEAT" (/HH
        ↪ EE T/).
each:
    Issue: "IY" was replaced with "IH", indicating a substitution error.
    Suggestion: Emphasize the initial /IY/ sound as in "EACH" (/IY CH/), ensuring the vowel is pronounced longer and
        ↪ with more tension.
other's:
    Issue: "DH" was replaced with a foreign-accented sound, indicating a substitution error. "ER" was replaced with
        ↪ "AH", indicating a substitution error.
    Suggestion: Practice the voiced consonant /DH/ as in "THIS" (/DH IH S/) ensuring the tongue is placed between the
        ↪ teeth. Practice /ER/ as in "HER" (/HH ER/), focusing on the rhotic r-coloring and avoiding vowel
        ↪ substitution.
face:
    Issue: "EY" was replaced with "EH", indicating a substitution error.
    Suggestion: Practice the /EY/ sound as in "FACE" (/F EY S/), distinguishing it from the shorter /EH/ vowel as in
        ↪ "FEST" (/F EH S T/).
```

(d) Whisper Large + Llama-3.1-8B:

```
the:
    Issue: "DH" was replaced with "D", indicating a substitution error.
    Suggestion: Practice the distinction between /DH/ and /D/ with pairs like "THY" (/DH AY/) vs. "DIE" (/D AY/).

stared:
    Issue: "R" was deleted, indicating a deletion error.
    Suggestion: Ensure the /R/ sound is pronounced by practicing words like "RED" (/R EH D/).

into:
    Issue: "IH" was replaced with "IY", indicating a substitution error.
    Suggestion: Practice the distinction between /IH/ and /IY/ with pairs like "BIT" (/B IH T/) vs. "BEET" (/B IY T/).

other's:
    Issue: "Z" was replaced with "S", indicating a substitution error.
    Suggestion: Practice the distinction between /Z/ and /S/ with pairs like "ZOO" (/Z UW/) vs. "SUE" (/S UW/).
```

Figure 11: Inference Output Examples