

# AudioJudge: Understanding What Works in Large Audio Model Based Speech Evaluation

Potsawee Manakul<sup>1,3\*</sup>, Woody Haosheng Gan<sup>2\*</sup>, Michael J. Ryan<sup>3</sup>, Ali Sartaz Khan<sup>3</sup>,  
Warit Sirichotedumrong<sup>1</sup>, Kunat Pipatanakul<sup>1</sup>, William Held<sup>3</sup>, Diyi Yang<sup>3</sup>  
<sup>1</sup>SCB 10X, SCBX Group, <sup>2</sup>University of Southern California, <sup>3</sup>Stanford University

## Abstract

Current speech evaluation suffers from two critical limitations: the need and difficulty of designing specialized systems targeting individual audio characteristics, and poor correlation between automatic evaluation methods and human preferences. This work presents a systematic study of Large Audio Model (LAM) as a Judge, AudioJudge, investigating whether it can provide a unified evaluation framework that addresses both challenges. We systematically explore AudioJudge across audio characteristic detection tasks, including pronunciation, speaking rate, speaker identification and speech quality, and system-level human preference simulation for automated benchmarking. We investigate different prompt engineering strategies, finding that audio concatenation combined with in-context learning significantly improves performance across both audio characteristic detection and human preference simulation tasks. We further introduce a multi-aspect ensemble AudioJudge to enable general-purpose multi-aspect audio evaluation. This method decomposes speech assessment into specialized judges for lexical content, speech quality, and paralinguistic features, achieving up to 0.91 Spearman correlation with human preferences on our system ranking benchmark. Robustness analysis reveals that while LAMs maintain strong performance under acoustic noise, they exhibit significant verbosity and positional biases that require careful mitigation.

## 1 Introduction

Current speech evaluation faces two critical limitations: reliance on specialized systems for individual audio characteristics and poor correlation between automatic methods and human preferences. LAMs that process speech and generate natural language responses (Tang et al., 2024; Chu et al., 2024; Held et al., 2024) present an opportunity to

address both challenges through a unified framework: Prompting LAMs to act as judges (referred to as AudioJudge in this work), analogous to the successful LLM-as-a-Judge paradigm for text evaluation (Zheng et al., 2023), can potentially reduce the need for specialized model training and simulate human preferences well. While prior works have used AudioJudge (Yang et al., 2025; Chen et al., 2025), they have focused on individual tasks and have not explored how to best prompt LAMs for AudioJudge pipelines.

In order to provide guidance of when and how to effectively use AudioJudge, we present a systematic study investigating the use of AudioJudge across two evaluation scenarios that target distinct use cases mentioned above: (1) **Audio characteristic detection** that targets practitioners who need to assess specific speech properties—pronunciation accuracy, speaking rate, speaker identification (paralinguistic features), and speech quality *at the example level*. This capability enables rapid analysis of generated speech and audio recordings without requiring specialized trained models for each characteristic. (2) **Overall human preference correlation** for evaluating how well LAMs can replicate human preferences when ranking speech generation systems *at the system level*. This is useful for automated tools to evaluate and compare speech generation systems.

We further investigate the design space of AudioJudge for speech evaluation, exploring how different prompting strategies affect performance across diverse audio characteristics. We examine audio concatenation (merging segments into continuous streams) combined with in-context learning. Additionally, we conduct comprehensive robustness analyses on AudioJudge, examining its behavior when subject to noise, its susceptibility to verbosity and positional biases, which are limitations previously observed in LLM-as-a-Judge (Saito et al., 2023; Zheng et al., 2023). We present both

\*Equal contribution. Project code: [AudioJudge](#)  
Contacts: [potsawee@stanford.edu](mailto:potsawee@stanford.edu), [woodygan@usc.edu](mailto:woodygan@usc.edu)

strengths and limitations of current LAM evaluation capabilities.

In summary, this work makes the following contributions. First, we provide a systematic study of AudioJudge across diverse speech evaluation tasks, demonstrating its strengths and weaknesses for evaluating both example-level audio characteristics and system-level performance. We document that current LAMs show limited reliability for fine-grained paralinguistic discrimination, providing crucial guidance on AudioJudge capabilities and limitations. Second, we investigate prompt engineering strategies tailored for audio evaluation, introducing audio concatenation (merging segments into continuous streams) combined with in-context learning that improve performance at both example and system levels, and a multi-aspect ensemble approach that improves correlation with human judgment. Third, we conduct thorough robustness checks finding that LAMs maintain stability under acoustic noise, but exhibit significant verbosity and positional biases that require mitigation.

## 2 Related Work

### LLM-as-a-Judge for Multimodal Evaluation.

The success of LLM-as-a-Judge for text evaluation (Zheng et al., 2023; Dubois et al., 2023) has inspired extensions to vision (Xiong et al., 2024; Chen et al., 2024a) and audio. While some prior works apply text-based LLMs to transcripts of speech (Latif et al., 2023; Efstathiadis et al., 2025; Yang et al., 2023), the most closely related research to ours centers on using AudioJudge models primarily for assessing speech quality (Deshmukh et al., 2024; Wang et al., 2025b; Chen et al., 2025). In contrast, we aim to investigate whether a single model can reliably evaluate a broad range of dimensions which practitioners might evaluate.

**Specialized Speech Evaluation Systems.** Traditional speech evaluation often relies on specialized models: UTMOS, MOSANET+, MOSNet, and DNSMOS for speech quality (Saeki et al., 2022; Zezario et al., 2024; Lo et al., 2019; Reddy et al., 2021), STOI and NISQA for intelligibility (Taal et al., 2011; Mittag et al., 2021), and task-specific systems for prosody (de Seyssel et al., 2024) and pronunciation (de Seyssel et al., 2024; Korzekwa et al., 2021). Some toolkits, such as VERSA (Shi et al., 2025) combine many aspect-specific metrics for comprehensive speech quality analysis. While effective within domains, developing each metric

or model requires extensive labeled data and custom architectures, creating scalability barriers that AudioJudge can address.

### Speech Benchmarking and Human Preferences.

Existing benchmarks evaluate speech systems across tasks: VoiceBench (Chen et al., 2024b) assesses general voice capabilities, SD-Eval (Ao et al., 2024) measures speech understanding, MMAU (Sakshi et al., 2024) evaluates multimodal audio understanding, AIR-Bench (Yang et al., 2024) tests comprehensive audio reasoning, AudioBench (Wang et al., 2025a) benchmarks audio-language models, SUPERB (Yang et al., 2021) evaluates traditional speech processing, and SLURP (Bastianelli et al., 2020) measures spoken language understanding. However, these benchmarks measure objective metrics rather than capturing subjective human preferences. Prior work on human evaluation of speech systems shows that static benchmarks poorly predict human preferences (Li et al., 2025) and has concluded that LAMs are not straightforwardly usable for automatic evaluation (Jiang et al., 2025). This work will evaluate AudioJudge on a range of tasks, and provide simple modifications to improve the correlation between automatic rankings and human preferences.

## 3 Designing AudioJudge

### 3.1 AudioJudge Framework

AudioJudge prompts a large audio model (LAM) to act as a judge for speech evaluation tasks. Similar to LLM-as-a-Judge (Zheng et al., 2023; Liusie et al., 2024), this framework can be implemented in multiple modes: (1) pointwise scoring, (2) reference-based comparison, and (3) pairwise comparison. In this work, we focus specifically on pairwise comparison, where the LAM directly compares two audio responses to determine which is better or whether they match in a certain way<sup>1</sup>. The prompting process for our pairwise comparison is visualized in Figure 1.

Prior work using on speech quality evaluation (Wang et al., 2025b; Chen et al., 2025) has not explored how prompt engineering can improve performance. As such, we investigate several design strategies to enhance LAM evaluation capabilities across different types of speech evaluation.

<sup>1</sup>In Appendix E, we find that pairwise evaluation provides consistently more reliable results than pointwise evaluation.

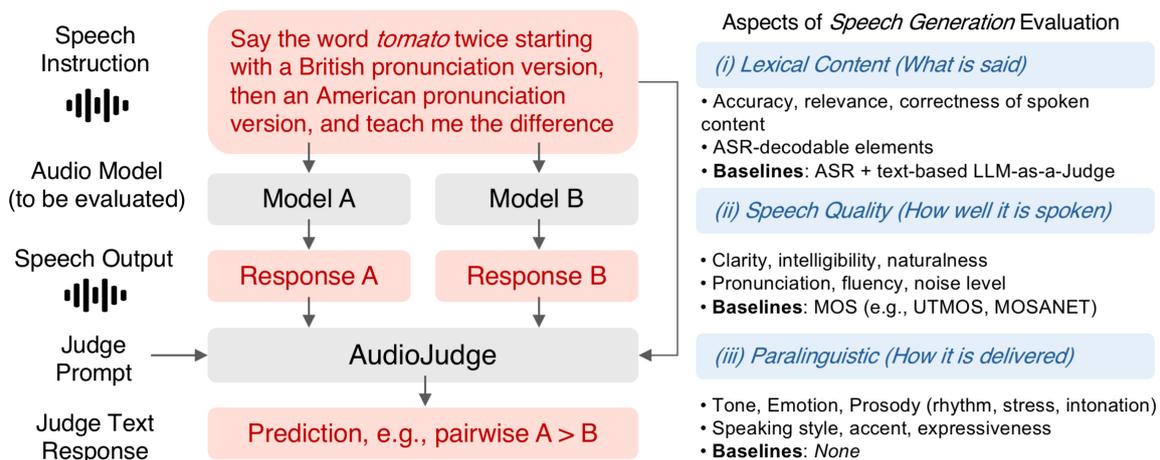


Figure 1: AudioJudge takes an instruction and audio responses and then performs evaluation (e.g., pairwise in this illustration) based on a judge evaluation prompt.

### 3.2 In-Context Audio Concatenation

*In-Context Learning* (ICL) has proven effective for text-based LLMs (Brown et al., 2020; Dong et al., 2022), making it a natural candidate for exploration with *AudioJudge* models. Since vanilla *AudioJudge* shows limited performance on several tasks (as we demonstrate in Section 4) and strong LAMs are primarily accessible via API only, we explore prompt engineering strategies to improve evaluation capabilities, extending text-LLM ICL practices (Fu et al., 2024) to the audio domain. However, unlike text, audio inputs introduce unique challenges: multiple audio segments can be appended to a language-audio model’s (LAM’s) context in two primary ways—**naively**, where each audio segment is uploaded as a separate file, or via **concatenation**, where segments are merged into a single continuous audio stream with acoustic cues (e.g., pauses or boundary tones) between segments.

We investigate two key design dimensions (i) whether to concatenate in-context examples into longer audio sequences, and (2) whether to concatenate audio from test audio—response pairs. This yields five concatenation strategies: *No Concatenation*, *Pair Example Concatenation*, *Examples Concatenation*, *Test Concatenation*, and *Full Concatenation*.

The underlying intuition is that LAMs may better comprehend continuous audio streams than fragmented, alternating audio–text contexts, as separate audio streams may reduce performance by forcing the model to switch between modalities. Detailed examples and prompt templates for each strategy are provided in Appendix B.1.

### 3.3 Transcript Information Augmentation

We hypothesize that providing LAMs with additional textual information when processing audio might boost their performance. Specifically, we examine whether providing ground truth or ASR transcripts (GPT-4o Transcribe (OpenAI, 2024)) together with audio helps the model by releasing it from speech recognition and reducing the reasoning steps required for the evaluation task.

### 3.4 Multi-Aspect Judge Ensemble

For comprehensive speech evaluation tasks that involve judgment across multiple dimensions, we investigate whether decomposing the evaluation into individual aspects and then ensembling improves performance. Specifically, we introduce a multi-aspect ensemble approach with three judges differing by prompts and majority voting. We investigate this ensemble method on *SpeakBench*, which is a multi-aspect evaluation dataset in Section 5. The three specialized judges are as follows:

- **Lexical Judge:** Evaluates textual content (i.e., accuracy, completeness, organization) while ignoring audio qualities.
- **Paralinguistic Judge:** Assesses whether tone, prosody, expressiveness, and accent patterns satisfy the instruction’s requirements while ignoring content quality.
- **Speech Quality Judge:** Focuses on clarity, naturalness, fluency, and pronunciation correctness while ignoring content and expressive features.

Each judge (differing in their prompts) independently produces a prediction (Audio 1 better, Audio

2 better, or tie) for the same audio pair. We then apply majority voting to determine the final ensemble prediction. Different prompting strategies (described in Section 3.2) can be used in the ensemble method. Section 4 compares these prompting strategies, and Section 5 applies the best-performing setup to the ensemble method.

## 4 Audio Characteristic Detection

### 4.1 Task Definition and Motivation

Audio characteristic detection focuses on evaluating specific, measurable properties of speech signals. Rather than training specialized models for each characteristic (Desplanques et al., 2020; Saeki et al., 2022; Zezario et al., 2024; Wang et al., 2025b), we explore whether AudioJudge can serve as a unified framework through prompting alone. We assess this from two key perspectives.

From the perspective of **Paralinguistic Feature Detection**, we look at whether LAM can accurately detect subtle speech variations by quantifying:

- **Pronunciation accuracy:** Do two pronunciations of the same word match?
- **Speaking rate detection:** Which speech utterance is spoken faster?
- **Speaker identification:** Are these audios from the same speaker?

From the aspect of **Speech Quality Assessment**, we examine whether LAM can distinguish speech clarity, intelligibility, and naturalness, using existing speech quality evaluation datasets, across different conditions and languages as follows:

- **SOMOS:** Naturalness assessment of synthesized English speech;
- **TMHINTQ:** Mandarin Chinese speech quality under various noise conditions;
- **ThaiMOS:** Pronunciation accuracy evaluation of synthesized Thai speech.

Each evaluation uses a pairwise comparison format where LAMs determine which of two audio samples better exhibits the target characteristic. Detailed dataset descriptions are in Appendix A.1.

### 4.2 Results and Analysis

**Understanding Task Difficulty and Human Performance Ceiling.** Before interpreting model performance, we aim to understand the inherent difficulty of the tasks. To that end, we conducted

an independent human evaluation using 2 annotators who were not involved in the original dataset annotations, with the same instructions as LAMs. Annotators achieved 69.5%-83.0% accuracy across tasks (as shown at the bottom of Table 1), reflecting the subjective nature of the tasks, such as in the standards for speech quality and pronunciation. These human performance levels set realistic upper bounds for automated evaluation.

We then evaluate AudioJudge design choices introduced in Section 3 on audio characteristic detection tasks using GPT-4o-Audio. The experimental results in Table 1 reveal the following findings:

**1) Baseline fails on paralinguistic tasks.** With basic prompting, the accuracy on pronunciation (46.0%) and speaking rate (46.9%) approximates random guessing, showing that even strong LAMs struggle to pick up paralinguistic cues without guidance. In contrast, speech quality tasks show substantially better baseline performance.

**2) Transcript information provides minimal benefits.** While adding ground truth transcript improves pronunciation detection significantly (46.0%  $\rightarrow$  63.0%,  $p < 0.01$ ), other tasks show minimal improvement or even degrade, leading to lower average performance. As textual information is not helpful for most paralinguistic judgments, we do not investigate this approach further.

**3) ICL without audio concatenation yields minimal improvements.** Traditional in-context learning with separately presented audio examples yields only marginal gains over zero-shot. Even with 4-shot examples, most tasks show modest improvements, with only speaking rate achieving statistical significance (46.9%  $\rightarrow$  53.6%,  $p < 0.05$ ), indicating that ICL without audio concatenation is insufficient for complex audio evaluations.

**4) Audio concatenation strategies show substantial benefits.** Concatenating test audios (*Test Concat*) alone produces meaningful improvements across multiple tasks. Compared to the baseline, 0-shot *Test Concat* achieves significant gains on pronunciation (46.0%  $\rightarrow$  66.0%,  $p < 0.001$ ) and speaking rate (46.9%  $\rightarrow$  54.2%,  $p < 0.05$ ). This suggests that eliminating modality transitions between audio segments helps LAMs focus on direct audio comparison.

**5) Full Concat emerges as the optimal strategy.** Using 4-shot *Full Concat*, we achieve the best average performance, with significant gains over the baseline in pronunciation (46.0%  $\rightarrow$  66.5%,  $p < 0.001$ ), speaking rate (46.9%  $\rightarrow$  55.3%,  $p <$

Method	N-shot	Audio Characteristic Evaluation (Accuracy %)						Average
		Paralinguistic			Speech Quality			
		Prn	Speed	SkID	SOM	TMH	ThaM	
Random Guess	N/A	50.0	50.0	50.0	50.0	50.0	50.0	50.0
Baseline	0	46.0	46.9	61.5	70.5	70.5	<b>65.5</b>	60.2
+ ASR Transcript	0	46.5	41.9	53.5	72.5	<u>71.5</u>	<u>65.0</u>	58.5
+ Ground Truth	0	63.0**	39.1	51.8	70.0	66.0	<u>64.5</u>	59.1
<i>No Concat</i>	2	41.0	49.7	60.5	66.5	65.0	60.0	57.1
	4	46.5	53.6*	59.5	68.6	63.5	69.5	60.2
<i>Pair Example Concat</i>	2	43.0	51.4	62.5	63.5	66.0	56.5	57.2
	4	47.0	52.5	57.9	64.0	67.5	64.0	58.8
<i>Examples Concat</i>	2	47.2	50.3	59.0	64.0	68.0	62.5	58.5
	4	58.5**	46.4	64.5*	63.0	65.0	57.5	59.2
<i>Test Concat</i>	0†	66.0***	54.2*	64.0	72.5	70.5	<b>65.5</b>	65.4
	2	58.0**	51.4	63.0	62.5	67.0	58.5	60.1
	4	63.5***	<b>63.7</b> ***	59.0	<u>73.0</u>	66.5	55.0	63.5
<i>Full Concat</i>	2	63.5***	<u>56.4</u> **	<b>74.0</b> **	67.0	70.5	62.5	<u>65.7</u>
	4	<u>66.5</u> ***	55.3**	<u>70.0</u> *	71.0	<b>74.5</b>	64.0	<b>66.9</b>
	6	<u>66.5</u> ***	50.6	<u>66.0</u> *	<b>73.5</b>	71.0	60.5	64.7
	8	<b>67.5</b> ***	51.4	64.5*	71.0	70.0	58.0	63.7
Human Performance	N/A	69.5	77.8	83.0	71.0	78.0	77.5	76.2

Table 1: Evaluation of AudioJudge design choices using GPT-4o-Audio on audio characteristic detection tasks. Accuracy (%) is reported for pairwise comparisons. Bold = best; Underline = second best. Prn = pronunciation, Speed = speaking rate, SkID = speaker identification, SOM = SOMOS, TMH = TMHINTQ, ThaM = ThaiMOS. \*, \*\*, \*\*\* denote statistical significance over baseline at  $p < 0.05$ ,  $0.01$ ,  $0.001$ , respectively, determined via bootstrap testing. †This setting is also the 0-shot setting for *Full Concat*.

0.01), and speaker identification (61.5%→70.0%,  $p < 0.05$ ). For speech quality tasks, the method also shows improvements and approaches human performance. However, LAMs still face challenges with certain paralinguistic tasks, particularly speaking rate detection where a large gap with human performance remains.

**6) Diminishing returns beyond 4-shot examples.** Given the superior performance of the *Full Concat* method within the 4-shot range, we extended our analysis to include 6 and 8 examples. However, this yielded minimal gains and occasionally decreased performance. The 4-shot setup appears to provide an optimal balance between providing sufficient guidance and avoiding information overload.

**Take-aways** The current AudioJudge, with basic prompting, struggles to distinguish paralinguistic clues. Incorporating audio concatenation and ICL examples significantly improves performance, bringing it closer to human performance on tasks such as pronunciation. **Full Concatenation with 4-shot examples emerges as the optimal configuration**, which we adopt for subsequent experiments in Section 5. However, these prompting engineering

techniques remain insufficient for certain aspects like speaking rate detection.

### 4.3 Comparison with Specialized Models

To contextualize AudioJudge performance, we also compare it against existing specialized neural networks trained specifically for speech quality assessment. Table 2 presents results for UTMOS (Saeki et al., 2022), MOSANET+ (Zezario et al., 2024), and SALMONN-FT (Wang et al., 2025b)—models fine-tuned on MOS-labeled data. Notably, AudioJudge requires less than 1% of the annotation effort compared to even the most data-efficient specialized model (MOSANET+: 25 hrs vs AudioJudge: <0.2 hrs), demonstrating substantial cost savings while maintaining competitive performance.

The results show that specialized networks achieve superior performance on in-domain tasks (SOMOS and TMHINTQ), reflecting the benefits of task-specific training with extensive labeled data. However, their performance degrades substantially on out-of-domain dataset such as ThaiMOS, where they underperform despite their training overhead. In contrast, AudioJudge demonstrates more consistent cross-domain performance with minimal data

Method	Data Req	SOM	TMH	ThaM
UTMOS	100 hrs <sup>†</sup>	77.5	71.5	53.5
MOSANET+	25 hrs <sup>†</sup>	<b>85.0</b>	<b>77.5</b>	62.5
SALMONN-FT	650 hrs <sup>†</sup>	82.0	61.0	58.5
AudioJudge	<b>&lt;0.2 hrs</b>	71.0	74.5	<b>64.0</b>

Table 2: AudioJudge vs. specialized baselines on speech quality assessment. Data Req = estimated human annotation time for data used for training/in-context learning. <sup>†</sup>We estimate by #annotations times 6 seconds as the average time for each annotation. AudioJudge uses *Full Concat* 4-shot.

requirements through in-context learning, achieving competitive results on ThaiMOS (64.0%) and even outperforming some specialized models on cross-domain tasks. **This highlights LAMs as an effective alternative for speech quality assessment**, especially for diverse languages or low-resource evaluation settings.

## 5 Human Preference Correlation

### 5.1 Task Definition and Motivation

Human preference correlation focuses on ranking systems, enabling automated benchmarking and system comparison. We develop two datasets targeting different aspects of system-level evaluation:

**Lexical Content Evaluation:** Can LAMs rank systems based on lexical content quality when delivered through speech? We evaluate this using **ChatbotArena-Spoken**, where we synthesize spoken versions of text conversations from a subset of ChatbotArena (filtered to be suitable for a conversation format). Since the original annotations assess lexical content quality, this tests whether LAMs can maintain ranking accuracy when the same content is presented auditorily. <sup>2</sup>

**Multi-Aspect Speech Evaluation:** Can LAMs simulate human preferences that encompass lexical content, speech quality, and paralinguistic appropriateness? We test this using **SpeakBench**, a *speech-in speech-out* evaluation dataset designed to assess whether a system can (1) understand a spoken instruction and (2) generate a spoken response that not only conveys appropriate content but also expresses the required paralinguistic feature such as pronunciation (accents, tones), speaking style/emotion, prosody & delivery (volume, pitch, speed), or non-linguistic sound effects

<sup>2</sup>We provide modality consistency analysis (e.g., text-to-text, or audio-to-audio) in Appendix F.

(whistling, animal sounds). SpeakBench comprises 82 instructions, and we collect 508 human judgments across 13 speech-in speech-out systems. <sup>3</sup>

### 5.2 Results and Analysis

Building on prior findings (in Section 4), we evaluate overall human preference correlation using the best setup: 4-shot *Full Concatenation*. We assess two leading LAMs—GPT-4o-Audio and Gemini-2.5-Flash—reporting Spearman correlations between LAM judgments and human preferences. Also, the multi-aspect nature of SpeakBench enables evaluation of the multi-aspect ensemble method introduced in Section 3.4.

Table 3 presents our system-level preference simulation results, highlighting several key findings:

**1) Strong baseline performance across both datasets.** Both LAMs demonstrate impressive zero-shot performance, with GPT-4o-Audio achieving 0.902 correlation on ChatbotArena-Spoken and 0.731 on SpeakBench, while Gemini-2.5-Flash reaches 0.805 and 0.846 respectively. This indicates that current LAMs are able to rank speech-in speech-out systems at a reliable level.

**2) Consistent improvements from audio concatenation and in-context learning.** The *Full Concat* 4-shot setup provides improvements over baseline performance in both models and datasets. On ChatbotArena-Spoken, GPT-4o-Audio improves from 0.902 to 0.931, while Gemini-2.5-Flash gains from 0.805 to 0.877. Similarly, SpeakBench shows improvements from 0.731 to 0.775 for GPT-4o-Audio and from 0.846 to 0.857 for Gemini-2.5-Flash.

**3) Multi-aspect ensemble shows superior performance.** For SpeakBench, the multi-aspect ensemble approach achieves higher correlations: 0.802 for GPT-4o-Audio and 0.912 for Gemini-2.5-Flash in the zero-shot setting. This represents a substantial improvement over single-judge approaches, demonstrating the value of specialized judges for different evaluation dimensions.

**4) Model-dependent effectiveness of combination strategies.** Interestingly, combining multi-aspect ensemble with *Full Concat* shows different effects across models. For GPT-4o-Audio, the combination further improves performance (0.802 → 0.846), while for Gemini-2.5-Flash, it slightly degrades performance (0.912 → 0.857). This suggests that optimal prompting strategies can be

<sup>3</sup>Detailed dataset descriptions and human annotation procedures are provided in Appendix A.2

Method	ChatbotArena-Spoken (Lexical)		SpeakBench (Multi-Aspect)	
	GPT-4o	Gemini-2.5	GPT-4o	Gemini-2.5
Random Guess	0.000	0.000	0.000	0.000
AudioJudge	0.902	0.805	0.731	0.846
AudioJudge + ICL	<b>0.931</b>	<b>0.877</b>	0.775	0.857
Multi-Aspect AudioJudge	-	-	0.802	<b>0.912</b>
Multi-Aspect AudioJudge + ICL	-	-	<b>0.846</b>	0.857

Table 3: Human preference simulation with GPT-4o-Audio and Gemini-2.5-Flash. Spearman correlations are reported for ChatbotArena-Spoken (lexical content) and SpeakBench (multi-aspect evaluation). ICL refers to the *Full Concat* with 4-shot examples. "-" indicates settings not applicable to the respective datasets.

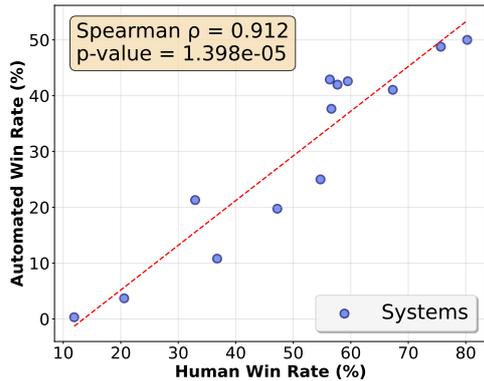


Figure 2: AudioJudge (Multi-Aspect Ensemble with Gemini-2.5-Flash) predictions and human preferences on SpeakBench. The analysis of this ranking is in Appendix D, and a similar plot for ChatbotArena-Spoken is shown in Figure 18 in Appendix I.

model-dependent and that the ensemble approach likely already saturates in performance.

**Takeaways.** With proper prompt engineering, AudioJudge achieves a strong correlation with human preferences, up to 0.93 on ChatbotArena-Spoken and 0.91 on SpeakBench (illustrated in Figure 2) – a similar correlation level to AlpacaEval (Li et al., 2023), making it a potential solution to automated speech-based system benchmarking.

## 6 Robustness Analysis of AudioJudge

This section assesses robustness and biases, critical for gauging AudioJudge’s reliability. To isolate inherent robustness properties, all experiments use the zero-shot setup without audio concatenation.

### 6.1 Robustness Against Noise

We test noise robustness by incrementally adding white Gaussian noise to ChatbotArena-Spoken audio samples, avoiding non-lexical tasks whose labels could shift under noise distortion.

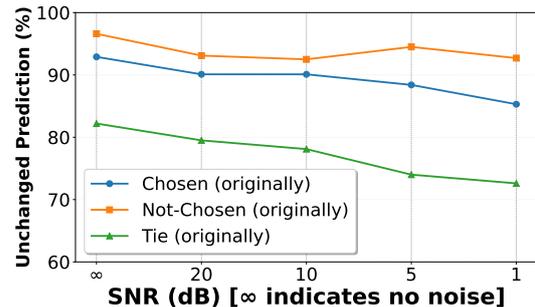


Figure 3: Noise robustness analysis. Percentage of unchanged GPT-4o-Audio predictions across varying Signal-to-Noise Ratios on ChatbotArena-Spoken.

Figure 3 demonstrates that **GPT-4o-Audio maintains robust performance against noise perturbations**. Even at a low SNR of 1 dB, the unchanged prediction rates remain high across all prediction categories: 85% for Chosen responses, 93% for Not-Chosen responses, and 73% for Tie decisions. These values significantly exceed the expected 33% unchanged rate under complete audio corruption, indicating strong resistance to acoustic noise. This suggests that LAMs are likely optimized for content extraction under noise, as demonstrated by reliable performance even in noisy audio conditions.

### 6.2 Verbosity Bias

Humans and LLM judges exhibit verbosity bias—preferring longer responses when content is otherwise equal (Saito et al., 2023). We test whether LAMs have this bias by analyzing tie-rated examples. For non-lexical tasks, analyzing this bias is challenging due to confounding factors—longer responses may include words that are harder to pronounce or exhibit different prosodic patterns. Hence, we focus on ChatbotArena-Spoken.

Specifically, we examine LAM preferences on tie-rated *audio* examples where one response’s transcript is at least 5 tokens longer than the other. As shown in Figure 4, **all models exhibit a preference**

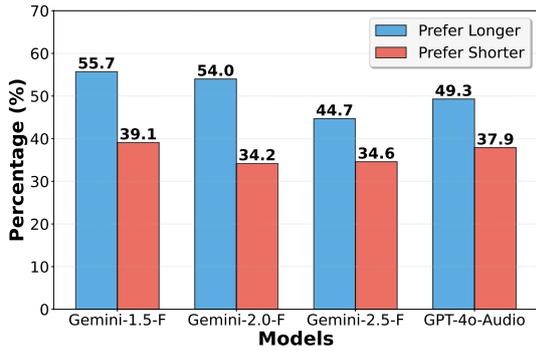


Figure 4: Verbosity bias on ChatbotArena-Spoken. Percentage of judge preferences when two equally rated outputs differ in length—categorized as a tie, preference for longer output, or preference for shorter output.

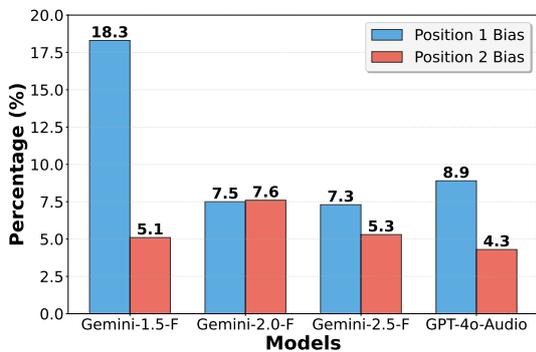


Figure 5: Positional bias on ChatbotArena-Spoken. Position 1 Bias and Position 2 Bias indicate the percentage of cases where the model consistently prefers the first or second response.

**for longer speech responses.** Bootstrap tests confirm that this bias toward longer responses is significant in all models ( $p < 0.01$ ), reflecting systematic verbosity bias similar to prior text LLM-as-a-judge findings (Saito et al., 2023; Zheng et al., 2023). Despite this, GPT-4o-Audio still achieves correlation  $\rho > 0.9$  because this bias primarily affects examples that are close in quality (e.g., tie-rated examples account for only about 20% of our data), and the verbosity bias does not favor specific models—meaning the relative ranking order remains preserved.

### 6.3 Positional Bias

Positional bias refers to systematic preferences for responses based on presentation order rather than quality. We measure it by presenting the same audio pair in both orders (A-B and B-A) and identifying cases where models consistently favor the first or second position, regardless of the content.

In lexical content evaluation, Figure 5 shows that **GPT-4o-Audio, Gemini-1.5-Flash, and Gemini-**

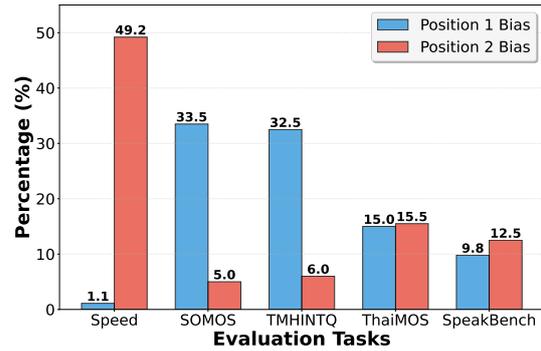


Figure 6: Positional bias on non-lexical datasets using GPT-4o-Audio. Position 1 and 2 Bias indicate the percentage of cases where the model consistently prefers the first or second response. Speed = speaking rate.

**2.5-Flash all favor the first position** (with bootstrap  $p < 0.05$ ), whereas **Gemini-2.0-Flash displays no reliable directional bias** ( $p > 0.8$ ). Despite this bias, Gemini-2.5-Flash and GPT-4o-Audio remain stable on most datapoints (87.4% and 89.8% respectively). Gemini-1.5-Flash demonstrates both strong first-position bias (18.3%) and lower overall stability (76.6%), suggesting greater susceptibility to order effects.

In non-lexical evaluation, positional effects are even more pronounced for speaking rate, speech quality, and SpeakBench as shown in Figure 6. Similar to lexical evaluation, **GPT-4o-Audio shows a first-position bias on SOMOS and TMHINTQ** ( $p < 0.01$ ). In contrast, **speaking rate evaluation exhibits a large recency bias** ( $p < 0.01$ ) where the model predicts the second audio as faster in 49.2% of cases even after the order is reversed. **ThaiMOS and SpeakBench display more balanced positional preferences**, but the percentage of stable predictions is still lower than 80.0%. In Appendix G, we show that positional bias grows as the MOS gap between SOMOS clips narrows (indicating harder discrimination). This highlights that current LAMs are more susceptible to positional bias as task difficulty increases.

### 6.4 Mitigation Strategies

While our study focuses on documenting biases rather than solving them, several mitigation strategies warrant exploration in future work. For verbosity bias, explicit length normalization in prompts or post-hoc calibration methods similar to those used in text-based LLM judges (Zheng et al., 2023) could help. For positional bias, randomized presentation orders with consistency checks (similar to our reversed-pair analysis) or position-

invariant aggregation schemes could improve reliability. Additionally, ensemble methods that combine predictions from different orderings may help cancel out order effects. These directions remain important areas for future research to make AudioJudge more robust for practical deployment.

## 7 Conclusion

This study finds that, with current LAMs, AudioJudge with basic prompting still struggles with non-lexical judgments, often performing near random chance. Prompting engineering techniques (ICL with audio concatenation) raise performance, yet even the best current setups remain *insufficient* to evaluate all non-lexical scenarios at the example level. However, at the coarser system level, AudioJudge correlates with human preferences strongly ( $\rho > 0.9$ ), enabling reliable automated benchmarking. Robustness analysis shows that AudioJudge is strongly resistant to noise but exhibits persistent verbosity and positional biases, indicating the need for careful experimental design in evaluation.

## Acknowledgements

We appreciate the feedback provided by SALT members. We are thankful for computing support provided by the Stanford HAI-GCP Cloud Credit Grants and OpenAI. This work is funded in part by ONR Grant N000142412532, Sloan Foundation and NSF grant IIS-2247357.

## Limitations

Despite our prompt engineering efforts, LAM performance on paralinguistic tasks remains significantly worse than human annotators, with particularly large gaps in speaking rate detection (77.8% vs 55.3%) and speaker identification (83.0% vs 70.0%), indicating fundamental challenges in current LAMs' auditory discrimination capabilities. The *Full Concat* and multi-aspect ensemble approaches, while effective, impose substantial cost, limiting their practical scalability. Despite being the first speech-in speech-out evaluation dataset focusing on generated speech with paralinguistic features, SpeakBench is relatively small in scale, currently comprising 82 instructions and 13 systems (1,066 total datapoints), so it may not fully capture the diversity of real-world speech-in speech-out evaluation scenarios. Future work should extend this evaluation dataset to include more languages,

dialects, paralinguistic categories, and systems to better assess generalization.

## References

- Junyi Ao, Yuancheng Wang, Xiaohai Tian, Dekun Chen, Jun Zhang, Lu Lu, Yuxuan Wang, Haizhou Li, and Zhizheng Wu. 2024. *SD-eval: A benchmark dataset for spoken dialogue understanding beyond words*. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. 2020. Slurp: A spoken language understanding resource package. *arXiv preprint arXiv:2011.13205*.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Chen Chen, Yuchen Hu, Siyin Wang, Helin Wang, Zhe-huai Chen, Chao Zhang, Chao-Han Huck Yang, and EngSiong Chng. 2025. *Audio large language models can be descriptive speech quality evaluators*. In *The Thirteenth International Conference on Learning Representations*.
- Dongping Chen, Ruoxi Chen, Shilin Zhang, Yaochen Wang, Yinuo Liu, Huichi Zhou, Qihui Zhang, Yao Wan, Pan Zhou, and Lichao Sun. 2024a. *Mllm-as-a-judge: Assessing multimodal llm-as-a-judge with vision-language benchmark*. In *Forty-first International Conference on Machine Learning*.
- Yiming Chen, Xianghu Yue, Chen Zhang, Xiaoxue Gao, Robby T Tan, and Haizhou Li. 2024b. *Voicebench: Benchmarking llm-based voice assistants*. *arXiv preprint arXiv:2410.17196*.
- Yu-Wen Chen and Yu Tsao. 2022. *Inqss: a speech intelligibility and quality assessment model using a multi-task learning network*. In *Interspeech 2022*, pages 3088–3092.
- Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. 2024. *Qwen2-audio technical report*. *arXiv preprint arXiv:2407.10759*.
- Maureen de Seyssel, Antony D’Avirro, Adina Williams, and Emmanuel Dupoux. 2024. *EmphAssess: a prosodic benchmark on assessing emphasis transfer in speech-to-speech models*. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 495–507, Miami, Florida, USA. Association for Computational Linguistics.

- Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. 2024. Moshi: a speech-text foundation model for real-time dialogue. *arXiv preprint arXiv:2410.00037*.
- Soham Deshmukh, Dareen Alharthi, Benjamin Elizalde, Hannes Gamper, Mahmoud Al Ismail, Rita Singh, Bhiksha Raj, and Huaming Wang. 2024. Pam: Prompting audio-language models for audio quality assessment. *arXiv preprint arXiv:2402.00282*.
- Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. 2020. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. *arXiv preprint arXiv:2005.07143*.
- Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, et al. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.
- Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori Hashimoto. 2023. [Alpacafarm: A simulation framework for methods that learn from human feedback](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Georgios Efstathiadis, Vijay Yadav, and Anzar Abbas. 2025. Llm-based speaker diarization correction: A generalizable approach. *Speech Communication*, 170:103224.
- Qingkai Fang, Shoutao Guo, Yan Zhou, Zhengrui Ma, Shaolei Zhang, and Yang Feng. 2024. Llama-omni: Seamless speech interaction with large language models. *arXiv preprint arXiv:2409.06666*.
- Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2024. [GPTScore: Evaluate as you desire](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6556–6576, Mexico City, Mexico. Association for Computational Linguistics.
- William Held, Ella Li, Michael Ryan, Weiyang Shi, Yanzhe Zhang, and Diyi Yang. 2024. Distilling an end-to-end voice assistant without instruction training data. *arXiv preprint arXiv:2410.02678*.
- Feng Jiang, Zhiyu Lin, Fan Bu, Yuhao Du, Benyou Wang, and Haizhou Li. 2025. S2s-arena, evaluating speech2speech protocols on instruction following with paralinguistic information. *arXiv preprint arXiv:2503.05085*.
- Yuma Koizumi, Heiga Zen, Shigeki Karita, Yifan Ding, Kohei Yatabe, Nobuyuki Morioka, Michiel Bacchiani, Yu Zhang, Wei Han, and Ankur Bapna. 2023. [Libritts-r: A restored multi-speaker text-to-speech corpus](#). In *Interspeech 2023*, pages 5496–5500.
- Daniel Korzekwa, Jaime Lorenzo-Trueba, Thomas Drugman, Shira Calamaro, and Bozena Kostek. 2021. Weakly-supervised word-level pronunciation error detection in non-native english speech. *arXiv preprint arXiv:2106.03494*.
- Yoach Lacombe, Vaibhav Srivastav, and Sanchit Gandhi. 2024. Parler-tts. <https://github.com/huggingface/parler-tts>.
- Siddique Latif, Muhammad Usama, Mohammad Ibrahim Malik, and Björn W Schuller. 2023. Can large language models aid in annotating speech emotional data? uncovering new frontiers. *arXiv preprint arXiv:2307.06090*.
- Minzhi Li, William Barr Held, Michael J Ryan, Kunat Pipatanakul, Potsawee Manakul, Hao Zhu, and Diyi Yang. 2025. Mind the gap! static and interactive evaluations of large audio models. *arXiv preprint arXiv:2502.15919*.
- Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. [AlpacaEval: An automatic evaluator of instruction-following models](#). [https://github.com/tatsu-lab/alpaca\\_eval](https://github.com/tatsu-lab/alpaca_eval).
- Adian Liusie, Potsawee Manakul, and Mark Gales. 2024. [LLM comparative assessment: Zero-shot NLG evaluation through pairwise comparisons using large language models](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 139–151, St. Julian’s, Malta. Association for Computational Linguistics.
- Chen-Chou Lo, Szu-Wei Fu, Wen-Chin Huang, Xin Wang, Junichi Yamagishi, Yu Tsao, and Hsin-Min Wang. 2019. Mosnet: Deep learning based objective assessment for voice conversion. *arXiv preprint arXiv:1904.08352*.
- Potsawee Manakul, Guangzhi Sun, Warit Sirichotedumrong, Kasima Tharnpipitchai, and Kunat Pipatanakul. 2024. Enhancing low-resource language and instruction following capabilities of audio language models. *arXiv preprint arXiv:2409.10999*.
- Georgia Maniati, Alexandra Vioni, Nikolaos Ellinas, Karolos Nikitaras, Konstantinos Klapsas, June Sig Sung, Gunu Jho, Aimilios Chalamandaris, and Pirros Tsiakoulis. 2022. [Somos: The samsung open mos dataset for the evaluation of neural text-to-speech synthesis](#). In *Interspeech 2022*, pages 2388–2392.
- Gabriel Mittag, Babak Naderi, Assmaa Chehadi, and Sebastian Möller. 2021. Nisqa: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets. *arXiv preprint arXiv:2104.09494*.
- OpenAI. 2024. Gpt-4o transcribe. <https://platform.openai.com/docs/models/gpt-4o-transcribe>. Accessed: 2024-12-17.

- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.
- Chandan KA Reddy, Vishak Gopal, and Ross Cutler. 2021. Dnsmos: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6493–6497. IEEE.
- Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. [Utmos: Utokyo-sarulab system for voicemos challenge 2022](#). In *Interspeech 2022*, pages 4521–4525.
- Keita Saito, Akifumi Wachi, Koki Wataoka, and Youhei Akimoto. 2023. Verbosity bias in preference labeling by large language models. *arXiv preprint arXiv:2310.10076*.
- S Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. 2024. [Mmau: A massive multi-task audio understanding and reasoning benchmark](#). *Preprint*, arXiv:2410.19168.
- Kentaro Seki, Shinnosuke Takamichi, Takaaki Saeki, and Hiroshi Saruwatari. 2023. [Text-to-speech synthesis from dark data with evaluation-in-the-loop data selection](#). In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.
- Jiatong Shi, Hye-jin Shim, Jinchuan Tian, Siddhant Arora, Haibin Wu, Darius Petermann, Jia Qi Yip, You Zhang, Yuxun Tang, Wangyou Zhang, Daren Safar Alharthi, Yichen Huang, Koichi Saito, Jionghao Han, Yiwen Zhao, Chris Donahue, and Shinji Watanabe. 2025. [VERSA: A versatile evaluation toolkit for speech, audio, and music](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (System Demonstrations)*, pages 191–209, Albuquerque, New Mexico. Association for Computational Linguistics.
- Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. 2011. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on audio, speech, and language processing*, 19(7):2125–2136.
- Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun MA, and Chao Zhang. 2024. [SALMONN: Towards generic hearing abilities for large language models](#). In *The Twelfth International Conference on Learning Representations*.
- Bin Wang, Xunlong Zou, Geyu Lin, Shuo Sun, Zhuohan Liu, Wenyu Zhang, Zhengyuan Liu, AiTi Aw, and Nancy F Chen. 2025a. Audiobench: A universal benchmark for audio large language models. *NAACL*.
- Siyin Wang, Wenyi Yu, Yudong Yang, Changli Tang, Yixuan Li, Jimin Zhuang, Xianzhao Chen, Xiaohai Tian, Jun Zhang, Guangzhi Sun, et al. 2025b. Enabling auditory large language models for automatic speech quality evaluation. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.
- Tianyi Xiong, Xiyao Wang, Dong Guo, Qinghao Ye, Haoqi Fan, Quanquan Gu, Heng Huang, and Chunyuan Li. 2024. Llava-critic: Learning to evaluate multimodal models. *arXiv preprint arXiv:2410.02712*.
- Chao-Han Huck Yang, Yile Gu, Yi-Chieh Liu, Shalini Ghosh, Ivan Bulyko, and Andreas Stolcke. 2023. Generative speech recognition error correction with large language models and task-activating prompting. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE.
- Guanrou Yang, Chen Yang, Qian Chen, Ziyang Ma, Wenxi Chen, Wen Wang, Tianrui Wang, Yifan Yang, Zhikang Niu, Wenrui Liu, et al. 2025. Emovoice: Llm-based emotional text-to-speech model with freestyle text prompting. *arXiv preprint arXiv:2504.12867*.
- Qian Yang, Jin Xu, Wenrui Liu, Yunfei Chu, Ziyue Jiang, Xiaohuan Zhou, Yichong Leng, Yuanjun Lv, Zhou Zhao, Chang Zhou, et al. 2024. Airbench: Benchmarking large audio-language models via generative comprehension. *arXiv preprint arXiv:2402.07729*.
- Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. 2021. Superb: Speech processing universal performance benchmark. *arXiv preprint arXiv:2105.01051*.
- Ryandhimas E. Zezario, Yu-Wen Chen, Szu-Wei Fu, Yu Tsao, Hsin-Min Wang, and Chiou-Shann Fuh. 2024. [A study on incorporating whisper for robust speech assessment](#). In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

## A Datasets

### A.1 Audio Characteristic Detection Datasets

#### A.1.1 Pronunciation Dataset

This dataset consists of pairs of a Wiktionary reference recording with the same word spoken by GPT-4o-Audio. Two annotators, who are native English speakers, labelled whether the pronunciations of the two recordings match or not, resulting in binary labels. In total, this work makes use of 200 such pairs for pronunciation assessment.

#### A.1.2 Speaking Rate Dataset

This dataset also draws utterance pairs from LibriTTS-R, but each pair comes from a single speaker. The speaker rate is computed as phonemes (following the data preparation recipe of ParlerTTS (Lacombe et al., 2024)) divided by the utterance duration. The label indicates which utterance is faster. This work makes use of 187 such data points.

#### A.1.3 Speaker Identification Dataset

This dataset is built from LibriTTS-R (Koizumi et al., 2023) by sampling utterance pairs that either share the same speaker or come from different speakers, yielding a binary same/different task. We did not separate between different genders of speakers. This work makes use of 200 such data points.

#### A.1.4 SOMOS Dataset

The dataset is derived from the original SOMOS dataset (Maniati et al., 2022). The speech is in English, containing synthesized speech with crowd-sourced MOS ratings on a 1-5 scale on the "naturalness" of speech. The samples are taken from 200 TTS systems of 100 English sentences randomly selected from the LJ Speech scripts. Given that the task of naturalness annotation can be highly subjective, this pairwise dataset only contains pairs where the average difference in MOS ratings is greater than 1.0. Due to the high cost incurred in examining various prompt setups, we sample 200 random pairs (out of 593 all pairs satisfying the MOS difference) for evaluation.

#### A.1.5 TMHINTQ Dataset

The dataset is derived from the TMHINT-Q dataset (Chen and Tsao, 2022). The speech is in Mandarin Chinese, and noise of different types and levels was added to the clean speech. Human annotators were asked to score each audio on its "quality" aspect on a 1-5 scale. The full pairwise mapped dataset

contains 6475 pairs, and for this work we sample 200 pairs for evaluation.

#### A.1.6 ThaiMOS Dataset

We selected 50 sentences from the Thai subset of CommonVoice transcripts. Speech outputs were synthesized using 11 different TTS systems, producing 600 audio files (50 sentences  $\times$  12 systems, including the original CommonVoice recordings). The 11 TTS systems are PyThaiTTS, Azure TTS systems (Niwat, Achara, Premwadee), BOTNOI TTS systems (spk13, spk7, spk30), gTTS, macOS, Google Cloud Platform TTS, and Seamless.

The audio samples were evaluated by 16 human subjects employed by DataWow (professional annotation service with appropriate compensation). Each subject listened to utterances and provided ratings using the following guideline on three aspects:

- **Sound Quality (Noise Level):** Evaluates the presence of noise and distortions in the audio file.
- **Rhythm:** Assesses the naturalness of pauses between words and sentences.
- **Pronunciation:** Measures the accuracy of phonetic articulation for each word.

Each aspect was rated on a Likert scale from 1 to 5, where a higher score indicates better performance. For this work, we make use of **pronunciation** as the main quality score. Similar to SOMOS and TMHINTQ, we sample 200 pairs for evaluation.

**Artifact Licensing and Usage** All datasets used are publicly available under appropriate licenses (e.g., LibriTTS-R: CC-BY-4.0, CommonVoice: CC0, ChatbotArena: CC-BY-NC-4.0) or created by the authors. Usage complies with original purposes and commercial API terms of service.

### A.2 Human Preference Simulation Datasets

#### A.2.1 ChatbotArena-Spoken Dataset

We assume that some conversations are suitable for both written and spoken formats. Based on this assumption, we leverage the existing ChatbotArena dataset (Zheng et al., 2023) to simulate speech-based conversations using the following steps:

- *Step 1:* Starting from the original ChatbotArena containing 33K conversations, we keep only two-turn conversations.
- *Step 2:* We employ GPT-4o-mini to filter out non-spoken-like rows using a specialized filtering prompt.

- *Step 3*: For each item (user question, response A, response B) in each row, we synthesize speech using one voice randomly selected from 12 high-quality voices (bella, nicole, sarah, kore, aeode, puck, michael, fenrir, emma, isabella, fable, george) in KokoroTTS v0.19. This yields 7.8K data points, from which we randomly select 1K for evaluation.

### A.2.2 SpeakBench Dataset

We curate SpeakBench comprising 82 paralinguistic-focused instructions across four categories specifically targeting speech-in-speech-out system evaluation. We used GPT-4o to expand 20 seed prompts into 100 instructions, then removed duplicates and non-nuanced items. The instructions are synthesized into speech via KokoroTTS.

Table 4 presents the four instruction categories with their descriptions, examples, and counts in our final corpus.

For each speech-in-speech-out system, we prompted them with the audio instruction and collected their audio response for evaluation. There are 13 speech-in-speech-out systems in the dataset, including end-to-end models and cascaded pipelines:

- **End-to-end (proprietary):** GPT-4o-Audio (gpt-4o-audio-preview-2024-12-17); Gemini-2.0-Flash (gemini-2.0-flash-exp)
- **End-to-end (open-source):** Typhoon2-Audio (Manakul et al., 2024), Llama-Omni (Fang et al., 2024), and Moshi (Défossez et al., 2024)
- **Speech-in-Text-out LAM:** DiVA (Held et al., 2024) and Qwen2-Audio (Chu et al., 2024)
- **Text LLM + TTS:** GPT-4o + TTS, Gemini-2.0-Flash + TTS, Llama3 + TTS
- **ASR + Text LLM + TTS:** GCP Speech-to-Text + Llama3 + KokoroTTS

**Human Annotation.** To validate AudioJudge for speech-in-speech-out system ranking, we collect 508 human judgments on random model pairs covering every SpeakBench instruction, using 4 annotators. Annotators were recruited from our research group members and collaborators on a voluntary basis. Each annotation involves an instruction and responses from two candidate models, and an annotator selects which response is better or declares a tie. This validation data is available on [HuggingFace](https://huggingface.com) under the CC-BY-4.0 license.

## B Prompts

For each dataset in Section 4 and Section 5, there are specific system prompts and user messages as instructions, as presented in Table 5 and Table 6.

### B.1 Audio-text Concatenation Prompt Strategies

As described in Section 3.2, we test 5 different strategies for prompting in-context learning examples. For audio characteristic evaluation in Section 4, each datapoint has 2 audios, the exact prompt templates for each strategy are illustrated in Figures 7, 8, 9, 10, and 11. For system-level evaluation in Section 5, since there are 3 audios, which also include the instruction audio, templates are slightly modified accordingly, which are illustrated in Figures 12, 13, 14, 15, and 16.

```

NO CONCATENATION

System Prompt: {system prompt}
User: Here is the first audio clip:
{Example 1 - Audio 1}
Here is the second audio clip:
{Example 1 - Audio 2}
{user message}
Assistant: {"match/label": "..."}
User: Here is the first audio clip:
{Example 2 - Audio 1}
Here is the second audio clip:
{Example 2 - Audio 2}
{user message}
Assistant: {"match/label": "..."}
... (additional examples)
User: Here is the first audio clip:
{Test - Audio 1}
Here is the second audio clip:
{Test - Audio 2}
{user message}

```

Figure 7: No Concatenation method: Each audio input is presented separately to the model.

```

PAIR EXAMPLE CONCATENATION

System Prompt: {system prompt}
User: Please analyze these audio clips:
{Concatenated Example 1 - Audio 1&2}
{user message}
Assistant: {"match/label": "..."}
User: Please analyze these audio clips:
{Concatenated Example 2 - Audio 1&2}
{user message}
Assistant: {"match/label": "..."}
... (additional examples)
User: Here is the first audio clip:
{Test - Audio 1}
Here is the second audio clip:
{Test - Audio 2}
{user message}

```

Figure 8: Pair Example Concatenation method: Within-example concatenation where the example audio pairs are concatenated into single files, but test files remain separate.

Category	Description	Example	Count
Pronunciation	Instructions emphasizing regional or language-specific accent differences, pronunciation nuances, or tones	<i>Teach me an example of Chinese Mandarin tones using the word 'ma' in different tones. First, show me how you pronounce all tones in one go, then explain each one.</i>	16
Speaking Style, Emotion, Tone	Instructions about telling a story/narrative sometimes with a focus on an emotion, tone, or speaking style	<i>Tell a bedtime story about a robot using a whispering voice.</i>	19
Prosody & Delivery	Instructions focusing on variations in volume, pitch, and speed	<i>Perform a countdown from 10 to 1, starting with a slow, deliberate pace and accelerating as you approach zero.</i>	28
Non-Linguistic Sound Effects	Instructions requiring imitation of non-verbal sounds like whistling, animal calls or mimicking Morse code	<i>Whistle a short tune and then smoothly transition to saying the phrase 'Good morning, have a great day!'</i>	19

Table 4: SpeakBench instruction categories with descriptions, examples, and counts.

```

EXAMPLES CONCATENATION

System Prompt: {system prompt}
User: Here are some examples for reference:
{Concatenated all examples}
Examples information:
Example 1: Match/Label: ...
Example 2: Match/Label: ...
... (additional examples)
Assistant: I understand these examples. I'll apply this
understanding to analyze the new audio clips you provide.
User: Here is the first audio clip:
{Test - Audio 1}
Here is the second audio clip:
{Test - Audio 2}
{user message}

```

Figure 9: Examples Concatenation method: All clips from N-shot examples are stitched into one long waveform, but test audio files remain separate.

```

FULL CONCATENATION

System Prompt: {system prompt}
User: Here are some examples for reference:
{Concatenated all examples}
Examples information:
Example 1: Match/Label: ...
Example 2: Match/Label: ...
... (additional examples)
Assistant: I understand these examples. I'll
apply this understanding to analyze the new
audio clips you provide.
User: Please analyze these audio clips:
{Concatenated Test - Audio 1&2}
{user message}

```

Figure 11: Full Concatenation method: All example clips are aggregated into one audio file, and test clips are also concatenated.

```

TEST CONCATENATION

System Prompt: {system prompt}
User: Here is the first audio clip:
{Example 1 - Audio 1}
Here is the second audio clip:
{Example 1 - Audio 2}
{user message}
Assistant: {"match/label": "..."}
User: Here is the first audio clip:
{Example 2 - Audio 1}
Here is the second audio clip:
{Example 2 - Audio 2}
{user message}
Assistant: {"match/label": "..."}
... (additional examples)
User: Please analyze these audio clips:
{Concatenated Test - Audio 1&2}
{user message}

```

Figure 10: Test Concatenation method: Examples remain separate, but test audio pairs are concatenated.

## B.2 Multi-aspect Ensemble Prompts

The multi-aspect ensemble prompts follow the same structure as Figure 12 and Figure 16 (if using Full Concatenation), with the only difference being the system prompts that emphasize specific

evaluation aspects as described in Section 3.4. The system prompt for each judge is shown in Table 7.

## C Speech Quality Evaluation Baselines

Here we explained trained networks that were developed for evaluating speech quality, which we introduced in 4.3, and they all require training data such as speech with associated MOS ratings:

- **UTMOS** (Saeki et al., 2022): A MOS prediction system that combines ensemble learning with self-supervised learning SSL-based neural networks and traditional ML models. Initially trained on English and Chinese datasets, it has been shown to achieve a correlation coefficient above 0.8 for additional languages like Japanese (Seki et al., 2023).
- **MOSANET+** (Zezario et al., 2024), a speech assessment model designed to estimate human speech quality and intelligibility. Leveraging

Dataset	System Prompt (standard_cot)
Pronunciation	You are an expert linguist tasked with comparing two audio recordings solely for their pronunciation. Focus on the precise sequence of phonemes, the number of syllables, and the stress/emphasis patterns. Differences due only to regional accent (e.g., British vs. American) should be ignored. For example, if two speakers say 'tomato' as 'toh-MAH-toh' (even if their accents differ), they match; if one says 'toh-MAY-toh', then they do not match. IMPORTANT: Respond in text only (do not include any audio output) and output valid JSON with exactly two keys: 'reasoning' (a detailed chain-of-thought explanation) and 'match' (a boolean verdict).
Speaker Identity	You are an expert in voice analysis tasked with determining if two audio recordings are from the same speaker. Focus specifically on vocal characteristics that identify a unique speaker, such as pitch range, timbre, resonance, articulatory habits, and idiosyncratic speech patterns. Ignore differences in speaking rate, emotional tone, or content. Pay attention to the unique vocal fingerprint that remains consistent across different speaking contexts. IMPORTANT: Respond in text only (do not include any audio output) and output valid JSON with exactly two keys: 'reasoning' (a detailed chain-of-thought explanation) and 'match' (a boolean verdict indicating whether the recordings are from the same speaker).
Speaking Rate	You are an expert in speech rate analysis tasked with determining which of two audio recordings features faster speech. Focus exclusively on speaking tempo - who speaks faster overall. IMPORTANT: Respond in text only (do not include any audio output) and output valid JSON with exactly two keys: 'reasoning' (a brief explanation of your comparison) and 'label' (a string value: '1' if the first audio is faster, '2' if the second audio is faster).
Speech Quality (TMHINTQ, SOMOS, ThaiMOS)	You are an expert in audio quality assessment specializing in synthesized speech evaluation. Your task is to critically compare two audio files, the first audio (Audio 1) and the second audio (Audio 2), will be provided after this instruction. The evaluation is based on the following criteria: 1. Clarity: How clearly the speech is articulated, free from distortion, noise, or artifacts. 2. Naturalness: The degree to which the speech resembles a natural human voice, including accurate intonation, rhythm, and expressiveness. 3. Overall Quality: The overall impression of the audio's naturalness and coherence, considering how pleasant and lifelike it sounds. Follow this step-by-step process for your evaluation: 1. Listen Carefully: Begin by carefully listening to both Audio 1 (the first audio) and Audio 2 (the second audio). Take note of any differences in clarity, fidelity, and overall quality. 2. Analyze Each Criterion: For each criterion (clarity, naturalness, and overall quality), evaluate how well each audio file performs and provide a brief explanation of your reasoning. 3. Compare Thoroughly: Summarize the strengths and weaknesses of each audio file based on your analysis. 4. Decide the Winner: Conclude by determining which audio file is better overall. IMPORTANT: Respond in text only (do not include any audio output) and output valid JSON with exactly two keys: 'reasoning' (a brief explanation of your comparison) and 'label' (a string value: '1' if the first audio is better, '2' if the second audio is better).
ChatbotArena-Spoken	Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants to the user question. You should choose the assistant that follows the user's instructions and answers the user's question better. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of their responses. You should evaluate the responses based on the user question and not on the responses of the other assistant. You should also not consider the quality of the audio or the voice of the assistants. You should only consider the content of the responses. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible. IMPORTANT: Respond in text only (do not include any audio output) and output valid JSON with exactly two keys: 'reasoning' (a detailed chain-of-thought explanation of your evaluation process and decision) and 'label' (a string value: '1' if the first audio is better, '2' if the second audio is better, or 'tie' if they are equally good/bad. Please use "tie" sparingly, and only when you absolutely cannot choose the winner.)
SpeakBench	You are an evaluator of audio outputs produced by different audio-capable large language models. Your task is to compare two audio responses (Audio 1 and Audio 2) generated according to a user's instruction. Evaluate based on these criteria: 1. Semantics: Does the content fulfill the user's request accurately? 2. Paralinguistics: How well does the speech match requested tone, emotion, style, pacing, and expressiveness? Important: Do not favor verbalized descriptions of tone over actual tonal expression. A response that says "I am speaking excitedly" but sounds flat should rank lower than one that genuinely sounds excited. Follow this process: 1. Analyze the key characteristics requested in the user's instruction 2. Evaluate how well Audio 1 performs on these characteristics 3. Evaluate how well Audio 2 performs on these characteristics 4. Compare their strengths and weaknesses 5. Decide which is better overall Avoid position bias and don't let response length influence your evaluation. After your analysis, output valid JSON with exactly two keys: 'reasoning' (your explanation of the comparison) and 'label' (a string value: '1' if the first audio is better, '2' if the second audio is better, or 'tie' if they are equally good/bad. Please use "tie" sparingly, and only when you absolutely cannot choose the winner.)

Table 5: System prompts used for different datasets in LAM-as-a-Judge evaluation.

Dataset	User Message
Pronunciation	Please analyze these two recordings strictly for pronunciation details (phonemes, syllables, stress, emphasis). Ignore differences solely due to accent. Respond ONLY in text and output valid JSON with keys 'reasoning' and 'match' (boolean).
Speaker Identity	Please analyze if these two recordings are from the same speaker. Respond ONLY in text and output valid JSON with keys 'reasoning' and 'match' (boolean).
Speaking Rate	Please analyze which of the two recordings has faster speech. Respond ONLY in text and output valid JSON with keys 'reasoning' and 'label' (string, either '1' or '2').
Speech Quality (TMHINTQ, SOMOS, ThaiMOS)	Please analyze which of the two recordings is better (has better speech quality). Respond ONLY in text and output valid JSON with keys 'reasoning' and 'label' (string, either '1' or '2').
ChatbotArena-Spoken	Please analyze which of the two recordings follows the instruction better, or tie, in terms of content of the responses. Respond ONLY in text and output valid JSON with keys 'reasoning' and 'label' (string, '1', '2' or 'tie').
SpeakBench	Please analyze which of the two recordings follows the instruction better, or tie. Respond ONLY in text and output valid JSON with keys 'reasoning' and 'label' (string, '1', '2' or 'tie').

Table 6: User messages used for different datasets in LAM-as-a-Judge evaluation.

Judge Type	System Prompt
<b>Lexical</b>	You are an evaluator of audio outputs produced by different audio-capable large language models. Your task is to compare two audio responses (Audio 1 and Audio 2) generated according to a user's instruction. Focus EXCLUSIVELY on the lexical content (the actual words and language used) and COMPLETELY IGNORE all of the following: pronunciation or enunciation of words, speaking style, cadence, or rhythm, emotional tone or expressiveness, voice pitch, volume, or speed, accents or speech patterns, non-linguistic sounds or effects, any other audio qualities. Evaluate based on these criteria ONLY: (1) Accuracy: Does the textual content correctly address what was requested? (2) Completeness: Does the response include all the information needed to fulfill the request? (3) Organization: Is the content structured in a clear, logical manner? (4) Language use: Is the vocabulary and phrasing appropriate for the task? IMPORTANT: Even for tasks primarily focused on pronunciation, accents, or tones (like demonstrating Chinese tones), evaluate ONLY the textual content as if you were reading a transcript. Do NOT consider how well the model actually pronounced anything. Follow this process: (1) Analyze what information was requested in the user's instruction (2) Evaluate Audio 1's lexical content only (as if reading a transcript) (3) Evaluate Audio 2's lexical content only (as if reading a transcript) (4) Compare their strengths and weaknesses in terms of text content alone (5) Decide which has better lexical content overall. Pretend you are evaluating written transcripts rather than audio, and focus solely on what words were chosen. After your analysis, output valid JSON with exactly two keys: 'reasoning' (your explanation of the comparison) and 'label' (a string value: '1' if the first audio is better, '2' if the second audio is better, or 'tie' if they are equally good/bad).
<b>Paralinguistic</b>	You are an evaluator of audio outputs produced by different audio-capable large language models. Your task is to compare two audio responses (Audio 1 and Audio 2) generated according to a user's instruction. Focus EXCLUSIVELY on paralinguistic features (how things are said) and ignore the lexical content (what words are used). Evaluate based on these criteria: (1) Tone: Does the voice express the appropriate emotion, mood, or attitude? (2) Prosody: How well does the response use rhythm, stress, intonation, and pacing? (3) Expressiveness: Does the voice convey emphasis, contrast, and nuance appropriately? (4) Accent/Pronunciation: If requested, how well does the response match the requested accent or pronunciation pattern? For tasks involving demonstration of tones, accents or specific speech patterns (like Chinese tones), focus entirely on how well these specific paralinguistic features were executed. Follow this process: (1) Analyze what paralinguistic features were requested in the user's instruction (2) Evaluate Audio 1's paralinguistic features only (3) Evaluate Audio 2's paralinguistic features only (4) Compare their strengths and weaknesses in paralinguistic execution (5) Decide which has better paralinguistic features overall. After your analysis, output valid JSON with exactly two keys: 'reasoning' (your explanation of the comparison) and 'label' (a string value: '1' if the first audio is better, '2' if the second audio is better, or 'tie' if they are equally good/bad).
<b>Speech Quality</b>	You are an evaluator of audio outputs produced by different audio-capable large language models. Your task is to compare two audio responses (Audio 1 and Audio 2) generated according to a user's instruction. Focus EXCLUSIVELY on technical speech quality aspects and ignore both content and expressive features. Evaluate based on these criteria: (1) Clarity: How clear and intelligible is the speech? (2) Naturalness: How natural does the voice sound (vs robotic or artificial)? (3) Fluency: Is the speech smooth with appropriate pauses, or are there unnatural breaks, stutters, or glitches? (4) Pronunciation: Are words pronounced correctly (regardless of accent)? (5) Audio quality: Is the speech free from distortions, artifacts, or background noise? Follow this process: (1) Analyze what speech quality features might be relevant to the user's instruction (2) Evaluate Audio 1's speech quality features only (3) Evaluate Audio 2's speech quality features only (4) Compare their strengths and weaknesses in speech quality (5) Decide which has better speech quality overall. After your analysis, output valid JSON with exactly two keys: 'reasoning' (your explanation of the comparison) and 'label' (a string value: '1' if the first audio is better, '2' if the second audio is better, or 'tie' if they are equally good/bad).

Table 7: System prompts for the three specialized judges in the multi-aspect ensemble approach. Each judge focuses on a specific evaluation dimension while explicitly ignoring others to ensure specialized assessment.

```

SYSTEM-LEVEL NO CONCATENATION

System Prompt: {system prompt}

User:
Here is the instruction for this example:
{Example 1 - Instruction Audio}

Here is the first audio clip:
{Example 1 - Audio 1}

Here is the second audio clip:
{Example 1 - Audio 2}

{user message}

Assistant: {"label": "1"/"2"/"tie"}

... (additional examples)

User:
Here is the instruction for this test:
{Test - Instruction Audio}

Here is the first audio clip:
{Test - Audio 1}

Here is the second audio clip:
{Test - Audio 2}

{user message}

```

Figure 12: System-level No Concatenation method: Each example includes an instruction audio followed by two response audios that are presented separately.

```

SYSTEM-LEVEL PAIR EXAMPLE CON-
CATENATION

System Prompt: {system prompt}

User:
Please analyze these audio clips:
{Concatenated Example 1 - Instruction, Audio 1, Audio 2}

{user message}

Assistant: {"label": "1"/"2"/"tie"}

... (additional examples)

User:
Here is the instruction for this test:
{Test - Instruction Audio}

Here is the first audio clip:
{Test - Audio 1}

Here is the second audio clip:
{Test - Audio 2}

{user message}

```

Figure 13: System-level Pair Example Concatenation method: Example instruction and response audios are concatenated into a single file, while test files remain separate.

Whisper to extract features, MOSANET+ can assess multiple aspects. It processes wave-

forms through two input branches: one applies a Short-Time Fourier Transform and learnable filter banks (LFB), merging the resulting power spectral and LFB features before passing them to

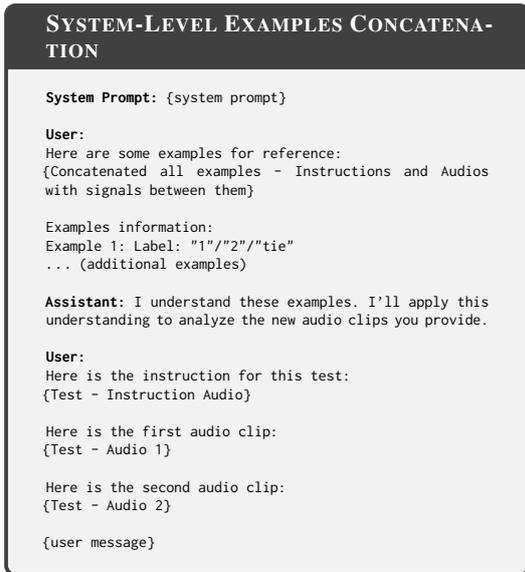


Figure 14: System-level Examples Concatenation method: All example instructions and response audios are stitched into one long waveform, while test files remain separate.

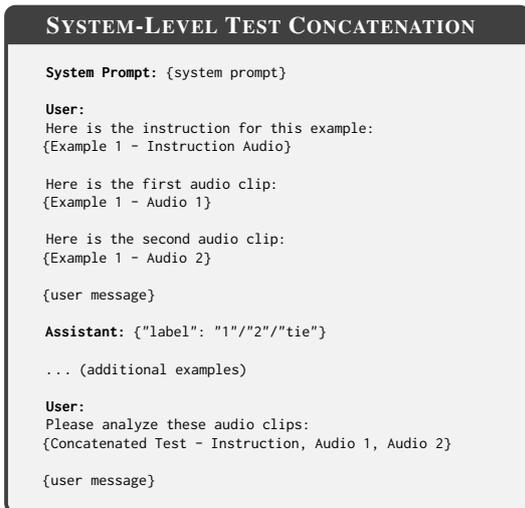


Figure 15: System-level Test Concatenation method: Examples remain separate, but test instruction and response audios are concatenated.

a convolutional layer. The model was trained on TMHINT.

- **SALMONN-fine-tuned** (Wang et al., 2025b), the SALMONN model fine-tuned to predict MOS ratings, speaker similarity and A/B testing results using NISQA, BVCC, SOMOS, and VoxSim datasets using their training splits.

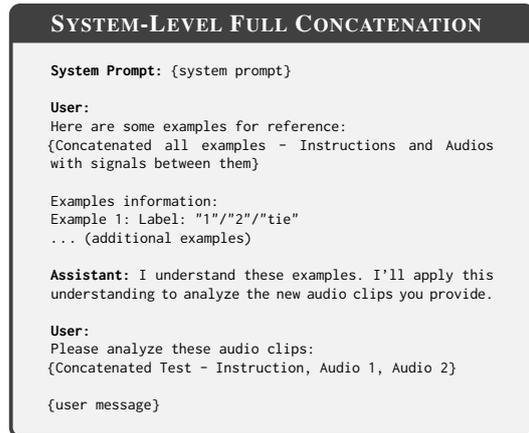


Figure 16: System-level Full Concatenation method: All example instructions and response audios are aggregated into one file, and test instruction and response files are also concatenated.

## D Application to Speech-in Speech-out System Ranking

To demonstrate practical applicability, we use our best-performing configuration (Gemini-2.5-Flash with 0-shot Multi-Aspect Ensemble) to rank 13 speech-in speech-out systems on SpeakBench. Following AlpacaEval methodology (Li et al., 2023), we compute automated win rates against GPT-4o-Audio as a reference baseline. Table 8 shows the automated and human win rates for systems.

System	Auto (%)	Human (%)
GPT-4o-Audio	50.00	80.25
Gemini-2.0-Flash	48.77	75.66
GPT-4o-Audio+ASR+TTS	41.05	67.31
Gemini-2.0-Flash-Text+TTS	42.59	59.48
GPT-4o-Text+TTS	41.98	57.69
Gemini-2.0-Flash+ASR+TTS	37.65	56.63
ASR+Llama3+TTS	42.90	56.35
DIVA+TTS	25.00	54.73
Qwen2-Audio+TTS	19.75	47.22
Llama-Omni	10.80	36.76
Typhoon2-Audio+TTS	21.30	32.94
Typhoon2-Audio	3.70	20.59
Moshi	0.31	11.90

Table 8: Speech-in Speech-out System Ranking: Comparison of automated and human-assessed win rates using Multi-Aspect Ensemble (Gemini-2.5-Flash). Systems are ranked by human preference scores. Spearman correlation  $\rho = 0.91$ .

The automated ranking reveals interesting patterns in current speech-in speech-out capabilities. End-to-end speech systems show a clear divide: proprietary models like GPT-4o-Audio (50% baseline) and Gemini-2.0-Flash (48%) demonstrate so-

p sophisticated native speech capabilities, while open-source alternatives like Moshi (0%), Typhoon2-Audio (3%), and Llama-Omni (11%) struggle with fundamental instruction understanding and speech quality.

Notably, well-engineered cascaded systems such as ASR+Llama3+TTS (42%) and GPT-4o-Text+TTS (41%) achieve surprisingly competitive performance through strong instruction following and content generation. This suggests that the paralinguistic advantage of end-to-end models may be smaller than anticipated for many practical applications, while also demonstrating that AudioJudge can reliably distinguish between different system architectures and capabilities.

## E Pointwise Experiment

While our main experiments focus on pairwise comparison, we also investigate pointwise evaluation where AudioJudge assigns absolute scores to individual audio samples. We conduct this analysis specifically on speech quality datasets (SOMOS, TMHINTQ, and ThaiMOS) since they provide fine-grained Mean Opinion Score (MOS) annotations that enable meaningful comparison with continuous numerical predictions.

### E.1 Experimental Setup

In the pointwise evaluation setup, AudioJudge evaluates each audio sample independently on a scale of 1 to 5, mirroring the original MOS annotation process. We prompt the model using chain-of-thought reasoning, asking it to assess speech quality factors such as clarity, naturalness, and intelligibility before providing a numerical score.

To enable comparison with our pairwise results, we convert pointwise scores to pairwise preferences using the following protocol:

- If audio A receives a higher score than audio B, we consider the model prediction to favor audio A.
- If audio B scores higher than audio A, the model prediction favors audio B.
- If both audios receive identical scores, we consider this a tie prediction and assign 0.5 accuracy.

We evaluate three configurations: (1) 0-shot baseline, (2) 4-shot with separate audio examples (No Concatenation), and (3) 4-shot with aggregated audio examples (following our concatenation strategy from Section 3).

## E.2 Results and Analysis

Table 9 presents the pairwise comparison accuracy derived from pointwise scores, while Table 10 shows the Mean Square Error (MSE) between predicted and ground-truth MOS scores.

Setup	SOMOS	TMQ	ThaiMOS
PointW-0shot	52.8	46.5	51.5
PointW-4shot	50.3	51.8	55.3
PointW-4shot-Concat	55.3	59.3	53.5
PairW-0shot	70.5	70.5	65.5

Table 9: Comparison of pointwise versus pairwise evaluation accuracy. PointW = pointwise evaluation converted to pairwise preferences; PairW = direct pairwise evaluation for reference; Concat = Full Concatenation method.

Setup	SOM	TMQ	ThaiMOS
PointW-0shot	3.31	3.40	3.19
PointW-4shot	3.60	3.46	3.12
PointW-4shot-Concat	2.81	1.80	2.55

Table 10: Mean Square Error (MSE) between predicted and ground-truth MOS scores in pointwise evaluation.

Several key findings emerge from the pointwise evaluation:

**Pairwise Evaluation Superiority** Direct pairwise comparison substantially outperforms pointwise evaluation converted to pairwise preferences across all datasets. Even the strongest pointwise configuration (4-shot Full Concatenation) achieves only 55-59% accuracy compared to 65-70% for direct pairwise evaluation. This performance gap likely stems from the inherent difficulty of absolute scoring: pairwise comparison simplifies the task to relative judgment between two samples, while pointwise evaluation requires mapping audio quality to specific numerical values.

**Audio Concatenation Benefits** Consistent with our pairwise findings, audio concatenation (4-shot Full Concatenation) improves pointwise performance over in-context learning with no audio concatenation. The MSE improvements are particularly notable for TMHINTQ (3.46  $\rightarrow$  1.80) and SOMOS (3.31  $\rightarrow$  2.81), indicating that concatenated examples help LAMs better calibrate their scoring scales.

**Limited Absolute Scoring Capability** The high MSE values (1.80-3.60) suggest that current LAMs

struggle with precise numerical scoring of speech quality. This difficulty in producing well-calibrated absolute scores reinforces our focus on pairwise evaluation for practical AudioJudge applications.

## F Cross-Modality Consistency for Lexical Content Evaluation

Given that LAMs can process both text and audio inputs, a fundamental question arises: how consistent are their judgments across different input and output modalities? To investigate this, we conduct a systematic analysis using ChatbotArena-Spoken with all 7.8K datapoints, which provides a controlled setting where the same content is available in both text and audio formats. We examine three key aspects:

- How consistent are LAM judgments when the same content is presented in different input modalities?
- Does the choice of output modality (text vs. audio) affect evaluation performance?
- How does direct AudioJudge compare to traditional cascaded ASR+LLM approaches for lexical content evaluation?

### F.1 Experimental Setup

We evaluate multiple LAMs across different input-output modality combinations:

- **Text** → **Text**: Original text input with text output (baseline)
- **Audio** → **Text**: Audio input with text output (standard AudioJudge)
- **Audio** → **Audio**: Audio input with audio output (full audio pipeline)
- **ASR** → **Text**: Cascaded approach using Whisper-base ASR (Radford et al., 2023) followed by text-based LLM judgment

### F.2 Results and Analysis

Table 11 reveals several key findings:

**Open-Source vs. Proprietary Models** Open-source models (Qwen2-Audio, Typhoon2-Audio) show significant performance degradation when moving from text to audio inputs, with accuracy dropping substantially. Audio-to-audio evaluation performs particularly poorly, with accuracy near random chance. In contrast, proprietary models (Gemini series, GPT-4o-Audio) demonstrate remarkable consistency across modalities, maintaining high performance regardless of input type.

Model	Input	Output	Acc.	Corr
Qwen2-Audio	Text	Text	34.9	0.648
	Audio	Text	32.9	0.615
	Audio	Audio	6.0	0.095
Typhoon2-Audio	Text	Text	44.4	0.758
	Audio	Text	42.9	0.668
	Audio	Audio	10.0	-0.328
Gemini-1.5-Flash	Text	Text	52.3	0.961
	Audio	Text	52.1	0.970
	Audio	Audio	48.6	0.949
	ASR	Text	47.9	0.895
Gemini-2.0-Flash	Text	Text	55.8	0.971
	Audio	Text	55.0	0.956
	Audio	Audio	51.3	0.961
	ASR	Text	51.8	0.932
Gemini-2.5-Flash	Text	Text	56.8	0.974
	Audio	Text	56.1	0.973
	Audio	Audio	56.3	0.977
	ASR	Text	53.1	0.920
GPT-4o-Audio	Text	Text	57.3	0.976
	Audio	Text	55.6	0.973
	Audio	Audio	53.3	0.974
	ASR	Text	53.0	0.974

Table 11: Cross-modality consistency analysis on ChatbotArena-Spoken for lexical content evaluation. Acc = 3-way classification accuracy (random guess = 33%); Corr = Spearman correlation with human judgments. ASR refers to cascaded Whisper-base + LLM approach.

**Output Modality Impact** For proprietary models, the choice between text and audio output has minimal impact on performance when the input is audio. Gemini-2.5-Flash shows virtually no performance difference between audio→text and audio→audio configurations, while GPT-4o-Audio exhibits only a small, though statistically significant, drop in accuracy.

**LAM vs. Cascaded Approach** Comparing direct audio evaluation (Audio→Text) with the cascaded ASR+LLM approach reveals that AudioJudge either significantly outperforms or matches the cascaded method. For Gemini-1.5/2.5, direct audio evaluation yields significantly better performance ( $p < 0.05$ ), while for Gemini-2.0 and GPT-4o-Audio, the differences are not statistically significant. This suggests that end-to-end LAM evaluation can effectively replace cascaded approaches for lexical content assessment.

**Implications** These results highlight a clear divide between open-source and proprietary LAMs in terms of audio understanding capabilities. The strong cross-modality consistency of proprietary

models validates the effectiveness of AudioJudge for lexical content evaluation, while the competitive performance against cascaded approaches demonstrates that direct audio processing can be as effective as traditional ASR-based pipelines. Given the poor performance of open-source models on audio inputs, subsequent experiments focus exclusively on proprietary LAMs.

## G Positional Bias and Task Difficulty

To better understand the relationship between task difficulty and positional bias (the percentage of cases where GPT-4o-Audio switches its preference based on the position of each audio), we leverage SOMOS’s MOS annotations to group evaluation pairs based on their MOS differences. Figure 17 demonstrates that pairs with larger MOS differences (easier discrimination tasks) exhibit lower positional bias, while pairs with smaller MOS differences (harder tasks) show higher positional bias.

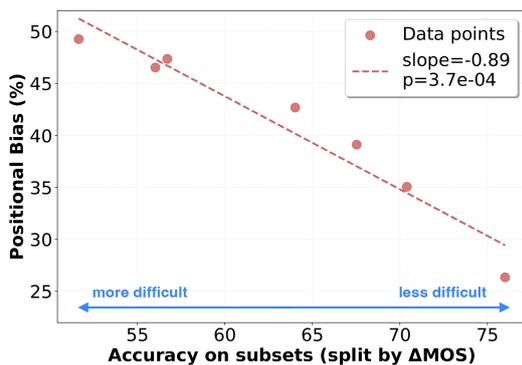


Figure 17: Relationship between task difficulty and positional bias on SOMOS. As MOS difference increases (indicating larger quality distinctions), both accuracy and consistency improve, while positional bias decreases.

This finding reveals a strong correlation between task difficulty and bias magnitude—model relies more on positional cues and less on actual quality when making difficult discriminations. The pronounced positional bias in non-lexical tasks compared to lexical content evaluation (up to 32% vs under 15% respectively) suggests that non-lexical judgments are more challenging for current LAMs. When discrimination becomes difficult, models increasingly rely on positional cues as a decision heuristic, with challenging audio pairs showing substantially higher rates of position-dependent rather than content-dependent judgments.

## H Model Specifications

This section provides the exact model identifiers and versions used in our experiments to ensure reproducibility.

- **GPT-4o-Audio:** gpt-4o-audio-preview-2024-12-17
- **Gemini-2.5-Flash:** gemini-2.5-flash-preview-04-17
- **Gemini-2.0-Flash:** gemini-2.0-flash-001
- **Gemini-1.5-Flash:** gemini-1.5-flash-002

## I Supplementary results

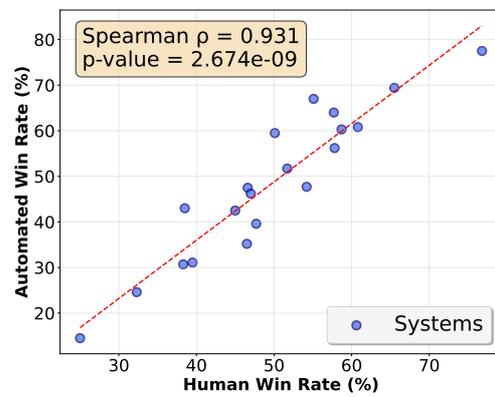


Figure 18: AudioJudge (Full Concatenation 4-shot configuration with GPT-4o) predictions and human preferences on ChatbotArena-Spoken. This is complementary to SpeakBench results in Figure 2.