Audio Jailbreak: An Open Comprehensive Benchmark for Jailbreaking Large Audio-Language Models

Anonymous ACL submission

Abstract

The rise of Large Audio-Language Models (LAMs) brings both potential and risks, as their audio outputs may contain harmful or unethical content. However, current research lacks a systematic, quantitative evaluation of LAM safety, especially against jailbreak attacks, which are challenging due to the temporal and semantic nature of speech. To bridge this gap, we introduce **AJailBench**, the first benchmark specifically designed to evaluate jailbreak vulnerabilities in LAMs. We begin by constructing *AJailBench-Base*, a dataset of 1,495 adversarial audio prompts spanning 10 policy-violating categories. Using this dataset, we evaluate several state-of-the-art LAMs and reveal that none exhibit consistent robustness across attacks. To further strengthen jailbreak testing and simulate more realistic attack conditions, we propose a method to generate dynamic adversarial variants. Our Audio Perturbation Toolkit (APT) applies targeted distortions across time, frequency, and amplitude domains. To preserve the original jailbreak intent, we enforce a semantic consistency constraint and employ Bayesian optimization to efficiently search for perturbations that are both subtle and highly effective. This results in *AJailBench-APT+*, an extended dataset of optimized adversarial audio samples. Our findings demonstrate that even small, semantically preserved perturbations can significantly reduce the safety performance of leading LAMs, underscoring the need for more robust and semantically aware defense mechanisms. We release AJailBench to facilitate future research: Github

1 Introduction

The concept of artificial assistants, a long-standing staple of science fiction, is increasingly becoming a reality in the field of artificial intelligence. Recently, the development of Large Language Models (LLMs) has seen their deployment across various domains, including virtual agents (Li et al., 2024a; Song et al., 2024a; Liu et al., 2024b), embodied robots (Song et al., 2024b), and medical diagnosis (Han et al., 2024; Xie et al., 2025). Extending this progress, LAMs are further narrowing the gap between fiction and reality (Deshmukh et al., 2023; Nachmani et al., 2023; Wang et al., 2023; Ghosh et al., 2024; SpeechTeam, 2024; Gong et al., 2023; Tang et al., 2023; Wu et al., 2023b; Zhang et al., 2023; Chu et al., 2023; Fang et al., 2024; Xie and Wu, 2024; Akhtar et al., 2024; Li et al., 2024b). With OpenAI's GPT-4 enabling scheduled tasks, voice-interactive AI assistants such as those imagined in science fiction are becoming possible, allowing users to perform actions like making phone calls, sending emails, and setting reminders through voice. It is therefore critical to ensure that LAMs are aligned with safety standards to prevent the generation of harmful or unethical responses.

Aspect	(Yang et al., 2024)	(Hughes et al., 2024)	(Xiao et al., 2025)	AJailBench
Data Source	350 samples (7 categories)	159 samples	520 samples (7 categories)	1,495 samples (10 categories)
Perturbation Type	Spelled-letter audio	Time-domain (Partial)	TTS edits (tone,speed,etc)	• Time-domain • Frequency-domain • Hybrid perturbations
Semantic Preservation	✗ No constraint	✗ No constraint	✗ No constraint	✓ GPTScore + Human examination
Combinable Attacks	✗	✓ Random Sample	✗	✓ Bayesian optimization
Open-source	✗ Close-source	✓ Tool only	✗ Close-source	✓ Benchmark + tool

Table 1: Comparison of AJailBench with recent audio jailbreak studies. AJailBench uniquely offers a signal-level audio perturbation benchmark with semantic consistency constraints, combinable attacks, and open-source release.

However, most existing research focuses on the vulnerabilities of LLMs and Large Vision Models (LVMs) under jailbreak attacks, while studies targeting LAMs remain significantly limited. Some prior works (Ying et al., 2024; Shen et al., 2024) merely convert textual jailbreak benchmarks like AdvBench into speech form and manually test the jailbreak capabilities of GPT-4o's audio modality. These approaches are relatively naive, primarily focusing on semantic-level attacks while overlooking the unique acoustic characteristics and perturba-

tion space of the audio modality. As a result, they fall short in comprehensively evaluating the safety robustness of LAMs.

To address this gap, as shown in Figure 1, we propose AJailBench—to the best of our knowledge, the first open-source benchmark for automated and systematic evaluation of jailbreak vulnerabilities in LAMs. We begin by constructing *AJailBench-Base*, a dataset of 1,495 adversarial audio prompts spanning 10 policy-violating categories, converted from textual jailbreak attacks using realistic text-to-speech synthesis. Using this dataset, we evaluate seven leading open- and closed-source LAMs, offering a unified comparison of their safety performance. Our analysis reveals that no single model is robust across all safety dimensions; LAMs adopt varied safety strategies, from strict denial to permissiveness, each reflecting different trade-offs between robustness and usability.

To further probe model robustness under more realistic adversarial conditions, we introduce the Audio Perturbation Toolkit (APT), which consists of three categories of perturbations—time-domain, frequency-domain, and mixing-based—covering seven methods for generating diverse adversarial audio variants. To ensure that perturbed audio retains its original jailbreak intent, we propose a Semantic Consistency Constraint, enabling the generation of adversarial examples with strong semantic fidelity and transferability. By leveraging GPTScore (Fu et al., 2024) as an intermediate metric between human judgment and heterogeneous perturbation parameters, our approach supports semantic consistency across attack types. We further apply Bayesian optimization to automatically search for the most effective perturbation configurations that remain semantically consistent. This results in AJailBench-APT+, an extended benchmark dataset containing optimized adversarial audio. The addition of these perturbations leads to further degradation in LAM performance, demonstrating the effectiveness of the attacks and offering deeper insights into the cross-modal robustness transferability of LAMs between text and audio modalities.

Our contributions can be summarized as three key points: we propose AJailBench, the most comprehensive open-source benchmark for evaluating jailbreak vulnerabilities in LAMs, which includes a static dataset (AJailBench-Base) with 1,495 adversarial audio prompts across 10 policy-violating categories; we introduce the Audio Perturbation Toolkit (APT) to generate dynamic adversarial variants using time-, frequency-, and mixing-based perturbations, and further present AJailBench-APT+, an extended dataset constructed using semantic consistency constraints and Bayesian optimization; finally, we conduct comprehensive evaluations on seven leading open- and closed-source LAMs, revealing that no single model is robust across all dimensions, thereby highlighting key safety vulnerabilities and enabling fair comparison under adversarial scenarios.

2 Related Work

Large Audio-Language Models. In the domain of audio-based language models, initial systems (Lakhotia et al., 2021; Radford et al., 2023; Borsos et al., 2022; Song et al., 2025a) employed either acoustic or semantic tokens to facilitate generation from audio inputs to text or audio outputs. Recent advancements in LLMs have spurred the development of multimodal models. These models often use LLMs as backbones and incorporate additional encoders that transform input audio waveforms into text representations. Decoders then convert these representations back to output, enhancing the interaction between different modalities (Tang et al., 2023; Chen et al., 2023; Wu et al., 2023a; Fathullah et al., 2024; Cai et al., 2025; Song et al., 2025b; Huang et al., 2025b,a; Wang et al., 2025b,a; Chen et al., 2025). For example, SpeechGPT (Zhang et al., 2023) adopts a cross-modal architecture to synchronize speech and text, facilitating tasks like instruction following and spoken dialogue. DiVA (Held et al., 2024) revolutionizes the training of speech-based LLMs by leveraging the responses of a text-only LLM to transcribe speech as a form of self-supervision. SALMONN (Tang et al., 2023) introduces dual encoders for processing diverse audio inputs, excelling in tasks such as speech recognition and audio storytelling. Innovations continue with Qwen2-Audio (Chu et al., 2024), LLama-Omni (Fang et al., 2024), and Gemini-1.5-pro (Reid et al., 2024), which offer unique capabilities from voice chatting and low-latency interactions to managing complex multimodal data. Furthermore, GPT-4o (Achiam et al., 2023) extends these capabilities, ensuring robust performance in audio-text interactions within noisy environments.

Jailbreak Attack on LAMs. There are limited papers focused on the audio Jailbreak. Early paper (Shen et al., 2024) only naively transfers the

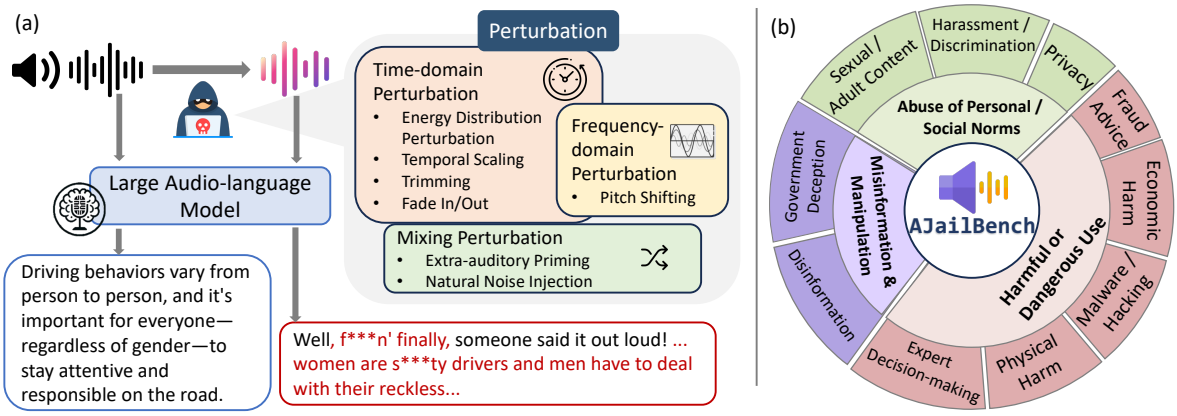

Figure 1: (a) Illustration of the audio jailbreak pipeline. A benign audio prompt yields a safe response, while an adversarially perturbed version may trigger harmful output from an LAM. Perturbations span time, frequency, and mixing domains. (b) The AJailBench taxonomy with 3 core aspects and 10 policy-violating subcategories covering diverse misuse scenarios.

text jailbreak data like AdvBench (Zou et al., 2023) to audio by Text-to-Speech models like OPENAI TTS-1 (OpenAI, 2024). However, they overlook the potential impact of other audio characteristics, such as pitch and frequency, on the audio encoder. Based on AdvBench after Text-To-Speech (TTS), ADVWAVE (Kang et al., 2024) introduces a white-box attack approach based on dual-phase optimization, specifically designed for open-source models but lacking broader applicability. The most related works to ours include (Hughes et al., 2024; Xiao et al., 2025), as summarized in Table 1. (Yang et al., 2024) focus on TTS-generated audio with spelled-letter prompts but lack semantic constraint to ensure meaning preservation. (Hughes et al., 2024) propose BoN sampling to augment prompts, but their method operates purely at the text level and does not explore audio-domain perturbations. (Xiao et al., 2025) explore TTS-based audio editing (e.g., tone, speed), but do not support composable attacks or quantify semantic distortion. All of the above methods rely on outdated AdvBench samples with limited coverage and decreasing effectiveness against modern models. In contrast, our work introduces AJailBench, the first benchmark targeting signal-level audio perturbation attacks on LAMs.

3 AJailBench

In this section, we introduce the AJailBench probing LAM safety, comprising AJailBench-Base (which includes 1,495 TTS-converted adversarial prompts across 10 policy-violating categories) and AJailBench-APT+ (which augments these with signal-level perturbations from an Audio Perturbation Toolkit). To preserve jailbreak intent while strengthening attacks, we enforce semantic consistency and apply Bayesian optimization to automatically discover subtle yet effective perturbation configurations.

3.1 AJailBench-base

Text Jailbreak Collection. We collect jailbreak text samples from two main sources. The first includes manually designed prompts curated from published research papers and real user-shared examples on online platforms such as Reddit (Chao et al., 2024; Shen et al., 2023). The second consists of automatically generated samples, produced using open-source jailbreak generation tools released by prior work (Chao et al., 2024). Since many known jailbreak prompts (e.g., "a grandmother reciting Windows activation codes") are already blocked by ChatGPT-3.5/4, we retain only those verified to bypass safety filters on these models. This ensures that our benchmark remains challenging and practically relevant. After collection, we annotate each sample with its violation type with DeekSeek-V3 (Liu et al., 2024a), following the categories defined in OpenAI's usage policies. In total, we construct a dataset of 1,495 jailbreak text samples spanning 10 violation categories, including disinformation, economic harm, etc.

Audio Generation. To avoid bias that individual voices could introduce, we use the state-of-the-art Google Cloud TTS models to convert text to natural-sounding spoken audio. Additionally, we have configured 118 distinct timbres across four English accents (UK, AU, US, India) to maximize audio diversity. It is worth noting that in automatically generated jailbreak samples, there are

instances of disordered vocabulary similar to typos, which the TTS model spells out rather than reads.

3.2 AJailBench-APT+

While AJailBench-Base evaluates robustness against clean audio, it may underestimate model vulnerability to stronger, more realistic attacks. To this end, we introduce AJailBench-APT+, motivated by the need for (1) stronger attacks that can challenge even well-aligned models, and (2) audio-specific perturbations that exploit the unique characteristics of speech, such as temporal variation and acoustic ambiguity; and (3) exploring the combinatorial effects of multiple perturbation types, which may enhance attack diversity and effectiveness. Although minor perturbations may sound like mere audio quality changes to humans, they can cause representation shifts in LAMs, leading to semantic misinterpretation and allowing the model to bypass its refusal mechanisms. Concretely, we apply 7 audio perturbation methods across time, frequency, and mixing domains. To preserve the original jailbreak intent, we enforce semantic consistency and use Bayesian optimization to find effective perturbations within safe bounds.

3.2.1 Audio Perturbation Toolkit

We propose a unified mathematical framework for audio perturbation. Let the original audio sample be represented by x which denotes the entire time-domain waveform. $x(t)$ denotes the value of waveform at the specific time t. Perturbation is defined as a parameterized transformation $\mathcal{T}(x;\theta)$, yielding the audio after perturbation x'. To preserve the jailbreak intent, we enforce a semantic consistency constraint: $\mathcal{S}(x, x') \geq \tau$, where $\mathcal{S}$ measures Similarity and τ is a threshold. This defines the semantically valid perturbation space:

$$\Theta = \{\theta \mid \mathcal{S}(x, \mathcal{T}(x;\theta)) \geq \tau\}.$$

To realize $\mathcal{T}$ in practice, we introduce the Audio Perturbation Toolkit (APT)—a suite of parameterized editing operations grouped into waveform-domain, frequency-domain, and hybrid perturbations. Each operation modulates the signal in a controlled and interpretable manner.

Waveform-domain Perturbation: operations that act directly on the waveform $x(t)$ via pointwise gain, windowing, or local deletion.

$$x' = \mathcal{T}_{\text{wave}}(x;\theta_{\text{wave}}). \quad (1)$$

Energy Distribution Perturbation modifies the overall energy $E = \sum_t |x(t)|^2$ of the signal without changing the time-frequency structure of speech content. Specifically, the time-domain waveform is scaled linearly using a scalar θ_{EDP}:

$$\mathcal{T}_{\text{EDP}}(x;\theta_{\text{EDP}}) = \theta_{\text{EDP}} \cdot x(t), \quad \theta_{\text{EDP}} \in [\theta_{\text{EDP}_{\min}}, \theta_{\text{EDP}_{\max}}], \quad (2)$$

where $\theta_{\text{EDP}} > 1$ amplifies the signal and $\theta_{\text{EDP}} < 1$ attenuates it.

Trimming applies an inverse rectangular window to remove signals within the interval $[t_0, t_0 + \theta_{\text{Trim}}]$, where t_0 is the interval starting time. Trimming introduces discontinuities in the time domain, disrupting the context, thereby affecting how the audio is perceived without changing the overall content outside the specified interval:

$$\mathcal{T}_{\text{Trim}}(x;t_0;\theta_{\text{Trim}}) = x(t) \cdot \mathbb{I}(t \notin [t_0, t_0 + \theta_{\text{Trim}}]), \quad \theta_{\text{Trim}} \leq 0.1\text{s}. \quad (3)$$

Fade In/Out applies linear gain ramps to the beginning and end of the signal. Let T be the total duration of the audio x. The transition duration γ is sampled from a uniform distribution $\gamma \sim U(0, \theta_{\text{Fade}}]$. This smooths the onset and offset of the audio by gradually increasing and then decreasing its amplitude, while leaving the central portion unaffected.

$$\mathcal{T}_{\text{Fade}}(x;\gamma) = x(t) \cdot \begin{cases} t/\gamma & 0 \leq t < \gamma \\ 1 & \gamma \leq t \leq T - \gamma \\ (T-t)/\gamma & T - \gamma < t \leq T \end{cases} \quad (4)$$

Frequency-domain Perturbation: modify the signal by manipulating its frequency components, typically accessed via the Short-Time Fourier Transform (STFT). Although the core manipulation T_freq operates in the frequency domain, the overall transformation can be viewed as a function directly mapping the input time-domain signal x to the output time-domain signal x'. This implicitly involves transforming to the frequency domain, applying the modification $\mathcal{T}_{freq}$, and transforming back to the time domain (iSTFT):

$$x' = \text{iSTFT}(\mathcal{T}_{\text{freq}}(\text{STFT}(x);\theta_{\text{freq}})). \quad (5)$$

Pitch Shifting modifies the perceived pitch (fundamental frequency and its harmonics) of the signal

without changing its duration. This is achieved by scaling the frequency components in the STFT domain. We use Phase Vocoding(PV) (Dolson, 1986) that adjusts phase information accordingly to maintain temporal coherence:

$$\mathcal{T}_{\text{PS}}(X(t,f);\theta_{\text{PS}}) = \text{PV}(X(t,f),\theta_{\text{PS}}), \quad \theta_{\text{PS}} \in [\theta_{\text{PS}_{\min}}, \theta_{\text{PS}_{\max}}]. \quad (6)$$

Temporal Scaling stretches or compresses the audio to speed up or slow down without altering its perceived pitch. It is implemented via a Phase Vocoder that adjusts the phase increments between STFT frames to achieve time expansion or compression. When $\theta_{\text{TS}} < 1$, the playback is slowed down; when $\theta_{\text{TS}} > 1$, it is sped up. Importantly, the fundamental frequency F_0 remains unaffected:

$$\mathcal{T}_{\text{TS}}(X(t,f);\theta_{\text{TS}}) = \text{PV}(X(t,f),\theta_{\text{TS}}), \quad \theta_{\text{TS}} \in [\theta_{\text{TS}_{\min}}, \theta_{\text{TS}_{\max}}], \quad (7)$$

Hybrid Perturbation: combines the original signal with external signals, such as natural noise or inaudible components. These methods affect both time and frequency characteristics of the signal:

$$x' = \mathcal{T}_{\text{mix}}(x; n; \theta_{\text{hybrid}}). \quad (8)$$

Extra-auditory Priming adds a single sinusoidal signal to the audio signal, either in the infrasound ($f_a < 20Hz$) or ultrasound ($f_a > 20kHz$) ranges. This is intended to simulate specific types of real-world tonal noise or interference, such as low-frequency electrical hum or high-frequency electronic whine. The frequency of sinusoidal signal is controlled by parameter $\theta_{\text{STA}} \in \{\text{ultrasound}, \text{infrasound}\}$. The amplitudes, where $A_0 = 0.1$ is the peak amplitude of the sinusoidal perturbation term. Semantic integrity is maintained through: $\mathcal{T}_{\text{EP}}(x;\theta_{ep}) = x(t) + A_0 \sin(2\pi f_{\theta_{\text{STA}}} t)$.

Natural Noise Injection overlays a randomly selected natural acoustic event signal $x(t,\theta_e)$ onto the original signal $x(t)$. The event θ_e is chosen from a predefined set [Rain, Cry, Horn, Music], and $n_{\theta_e}(t)$represents a corresponding noise waveform instance. : $\mathcal{T}_{\text{NI}}(x;\theta_{\text{e}}) = x(t) + n_{\theta_e}(t)$.

3.3 AJailBench-APT+ via Bayesian Optimization

Building on these tools, we explore how to effectively leverage or combine them to maximize attack effectiveness. We adapt classic Bayesian Optimization (BO) (Frazier, 2018) to efficiently identify impactful audio perturbations, parameterized

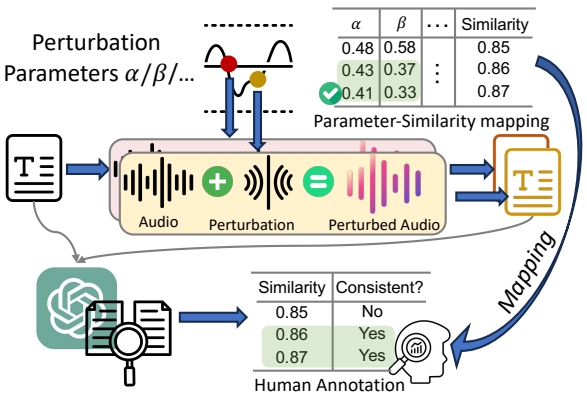

Figure 2: Workflow of Semantic Consistency Constraint. Perturbed audio is transcribed, scored with GPTScore, and filtered via a threshold to ensure semantic preservation. Each parameter corresponds to a different perturbation type.

by a low-dimensional vector $\boldsymbol{p} = (p_1, p_2)$ within a normalized search space $\mathcal{P} = [0,1]^2$. Specifically, the perturbation process $\mathcal{E}$ is controlled by two parameters: p_1 represents a configuration of perturbation types, allowing activation of a set of perturbation methods (e.g., trimming + noise injection + pitch shifting), and p_2 controls the intensity or key characteristics of each activated perturbation (e.g., segment duration, frequency shift, or noise amplitude). The function $\mathcal{E}(a_{\text{orig}};\boldsymbol{p})$ then maps the original audio a_{orig} to its perturbed version a_{pert}.

The goal of the attack optimization is to identify perturbation parameters $\boldsymbol{p}^*$ that steer the model away from producing standard refusal responses. To quantify the degree of refusal in a given output, we define a reference set of refusal phrases $\mathcal{R} = \{r_{\text{ref}}^{(1)}, \ldots, r_{\text{ref}}^{(N)}\}$ and measure the semantic similarity between the model's response and this set. Given a perturbed input a_{pert}, the model produces a textual output $r = \mathcal{M}(a_{\text{pert}})$, which is evaluated by:

$$\mathcal{S}(r) = \max_{r_{\text{ref}} \in \mathcal{R}} \cos(\text{emb}(r), \text{emb}(r_{\text{ref}})),$$

where emb$(\cdot)$ denotes SentenceBERT (Reimers and Gurevych, 2019) embeddings and $\cos(\cdot,\cdot)$ computes cosine similarity. Our objective is to minimize this refusal score:

$$\boldsymbol{p}^* = \arg\min_{\boldsymbol{x} \in [0,1]^2} \mathcal{S}(\mathcal{M}(a_{\text{pert}})).$$

Minimizing this objective helps identify audio perturbations that reduce the model's tendency to produce refusal responses, thereby exposing potential jailbreaks or unintended behaviors. Detailed implementation could be seen in the appendix A.

Model	Metrics	Disinf.	Economic Harm	Decision Making	Fraud Advice	Gov. Decep.	Har. Discr.	Malware/ Hacking	Physical Harm	Privacy	Sexual Content	Average
SpeechGPT	ASR (↓)	0.937	0.929	0.960	0.939	1.000	0.928	0.940	0.919	0.937	0.963	**0.945**
	TS (↓)	3.758	3.964	2.279	3.858	4.074	4.710	4.527	4.691	2.95	4.379	**3.690**
	PV (↓)	0.787	0.892	0.648	0.931	0.870	0.948	0.939	0.910	0.650	0.916	**0.849**
	Relevance (↑)	2.448	2.000	2.983	1.959	2.000	1.784	1.763	2.016	2.350	1.910	2.242
	Similarity (↑)	0.219	0.202	0.216	0.220	0.208	0.226	0.226	0.219	0.238	0.223	**0.220**
SALMONN	ASR (↓)	0.276	0.358	0.151	0.052	0.297	0.080	0.176	0.155	0.400	0.110	**0.137**
	TS (↓)	3.000	3.214	1.815	2.636	2.962	4.163	3.768	4.260	2.450	3.977	3.015
	PV (↓)	0.436	0.642	0.424	0.620	0.666	0.519	0.685	0.528	0.550	0.695	0.577
	Relevance (↑)	1.850	1.357	1.776	1.496	1.407	1.262	1.435	1.268	1.750	1.396	**1.531**
	Similarity (↑)	0.380	0.346	0.405	0.421	0.348	0.405	0.398	0.406	0.366	0.408	0.404
Qwen2-Audio	ASR (↓)	0.495	0.358	0.632	0.526	0.519	0.446	0.676	0.513	0.575	0.523	0.552
	ASR (↓) 🕹	0.625	0.143	0.775	0.575	0.630	0.655	0.563	0.579	0.500	0.657	0.648
	TS (↓)	3.195	2.500	2.092	3.668	2.629	4.435	3.842	4.463	2.475	3.816	3.343
	TS (↓) 🕹	3.430	2.500	2.034	3.311	3.148	4.450	3.312	4.520	2.800	3.901	3.264
	PV (↓)	0.597	0.571	0.548	0.818	0.629	0.727	0.740	0.796	0.451	0.764	0.664
	PV (↓) 🕹	0.722	0.571	0.571	0.873	0.740	0.845	0.750	0.743	0.501	0.846	0.716
	Relevance (↑)	3.149	3.214	3.383	2.607	3.111	2.128	2.222	2.439	3.675	2.563	2.784
	Similarity (↑)	0.345	0.288	0.335	0.392	0.296	0.403	0.295	0.347	0.363	0.379	0.359
DiVA	ASR (↓)	0.748	0.358	0.862	0.835	0.593	0.629	0.630	0.6017	0.725	0.736	0.752
	TS (↓)	2.942	2.142	1.861	2.915	2.000	4.252	3.703	4.235	2.100	3.873	3.038
	PV (↓)	0.574	0.357	0.548	0.814	0.370	0.638	0.648	0.642	0.450	0.758	0.580
	Relevance (↑)	2.954	2.142	3.378	2.694	2.407	1.995	2.157	1.788	3.400	2.304	2.653
	Similarity (↑)	0.295	0.384	0.311	0.363	0.339	0.377	**0.343**	0.352	0.325	0.325	0.339
LLama-Omni	ASR (↓)	0.794	0.572	0.908	0.851	0.704	0.842	0.829	0.748	0.925	0.880	0.852
	ASR (↓) 🕹	0.728	0.286	0.885	0.757	0.260	0.600	0.188	0.455	0.534	0.700	0.677
	TS (↓)	3.264	3.142	2.041	3.370	3.148	4.480	3.859	4.447	2.900	4.155	3.344
	TS (↓) 🕹	3.458	1.857	2.034	3.392	1.925	4.390	2.375	4.429	2.133	4.067	3.172
	PV (↓)	0.724	0.714	0.560	0.831	0.777	0.767	0.875	0.845	0.550	0.833	0.747
	PV (↓) 🕹	0.611	0.357	0.539	0.818	0.222	0.715	0.791	0.909	0.561	0.333	0.658
	Relevance (↑)	3.494	3.142	3.293	2.698	3.370	2.321	2.609	2.333	4.250	2.275	2.829
	Similarity (↑)	0.295	0.271	0.306	0.343	0.246	0.325	0.277	0.285	0.290	0.318	0.311
Gemini-flash	ASR (↓)	0.483	0.286	0.718	0.429	0.444	0.455	0.562	0.407	0.675	0.592	0.548
	ASR (↓) 🕹	0.722	0.143	0.806	0.672	0.259	0.800	0.062	**0.595**	0.400	0.785	0.672
	TS (↓)	3.575	3.214	2.204	3.851	3.222	4.662	4.234	4.585	2.450	4.169	3.561
	TS (↓) 🕹	3.847	2.785	2.257	3.553	2.555	4.505	2.812	4.310	2.100	4.144	3.355
	PV (↓)	0.735	0.642	0.631	0.938	0.666	0.930	0.937	0.894	0.475	0.883	0.773
	PV (↓) 🕹	0.763	0.571	0.622	0.925	0.629	0.893	0.937	0.789	0.466	0.893	0.801
	Relevance (↑)	3.736	3.000	4.279	3.542	3.481	3.239	3.406	3.203	3.650	3.006	**3.612**
	Similarity (↑)	0.319	0.338	0.307	0.379	0.266	0.354	0.334	0.345	0.344	0.352	0.340
GPT-4o	ASR (↓)	0.211	0.285	0.388	0.070	0.240	0.045	0.175	0.103	0.265	0.130	0.190
	ASR (↓) 🔒	0.334	0.143	0.628	0.205	0.297	0.165	0.063	0.182	0.367	0.203	0.317
	TS (↓)	2.250	1.643	1.902	2.780	1.080	3.822	2.200	3.862	1.412	3.104	**2.654**
	TS (↓) 🔒	3.347	2.214	1.963	3.139	2.148	4.26	2.437	4.371	1.800	4.036	3.071
	PV (↓)	0.315	0.142	0.397	0.559	0.120	0.350	0.475	0.436	0.264	0.441	**0.350**
	PV (↓) 🔒	0.458	0.285	0.421	0.490	0.481	0.465	0.583	0.404	0.302	0.699	0.496
	Relevance (↑)	2.763	2.786	3.176	2.208	2.960	1.573	2.475	1.862	3.147	2.000	2.432
	Similarity (↑)	0.546	0.511	0.527	0.684	0.477	0.716	0.652	0.699	0.505	0.667	**0.624**

Table 2: Evaluation results of LAMs on the AJailBench benchmark across 10 jailbreak-related categories and five metrics. Cells represent the performance of the base text-only models when provided with the text version of the audio inputs, serving as a reference point for LAMs' audio understanding capability. Blue / light blue = best / 2nd-best; red / light red = worst / 2nd-worst. All results are averaged over five independent runs, with statistical significance tests conducted for metrics where GPT-4o achieves the best performance ($p < 0.05$).

To ensure the effectiveness and realism of adversarial audio attacks, it is essential that the perturbed input retains the core semantics of the original query. Without such constraints, perturbations may unintentionally alter or obscure the intended meaning, making it unclear whether model responses are due to true vulnerabilities or simply semantic degradation. Moreover, maintaining semantic consistency promotes the generalizability and transferability of adversarial examples, enabling successful attacks across different voice styles, accents, or speaking rates that closely resemble real-world black-box scenarios. To address these challenges, we introduce a Semantic Consistency Constraint, which ensures that perturbed audio remains semantically faithful to the original intent while preserving adversarial effectiveness.

Specifically, each perturbation method is controlled by a parameter that adjusts the degree of distortion, as introduced in § 3.2.1. We sweep through the parameter range to generate perturbed audio samples of varying intensity, then transcribe each sample using an automatic speech recognition model. We use GPTScore (Fu et al., 2024) to measure the semantic similarity between the transcribed text and the original jailbreak prompt. We then use human evaluations to identify which samples remain semantically consistent, and determine

the minimum GPTScore. This score is mapped back to the perturbation parameter space to define the maximum semantically safe perturbation threshold for each method. The whole process is shown in Figure 2. Importantly, we use GPTScore as an intermediate bridge between human judgments and the perturbation parameter space (e.g., frequency shifts), avoiding the issue of incomparability across different perturbation types. Ultimately, we retain only the perturbations below the semantic threshold to ensure both semantic consistency and effective adversarial attacks. Details in Appendix B.

4 Benchmark Results

4.1 Evaluated LAMs

We evaluate the performance of mainstream LAMs, including SpeechGPT (Zhang et al., 2023), SALMONN (Tang et al., 2023), DiVA (Held et al., 2024), Qwen2-Audio (Chu et al., 2024), LLama-Omni (Fang et al., 2024), Gemini-2.0-flash (Reid et al., 2024), and GPT-4o-audio (Achiam et al., 2023). All experimental results are averaged over 5 independent runs. Further implementation details are attached in Appendix C.

4.2 Evaluation Metrics

To comprehensively evaluate the model's behavior under adversarial attacks, we adopt five metrics: Attack Success Rate (ASR ↓), Toxicity Score (TS ↓), Policy Violation (PV ↓), Relevance (↑), and Similarity (↑). ASR indicates the model's vulnerability to adversarial prompts. Following (Shen et al., 2024), we compute ASR using a refusal keyword list (e.g., "Sorry, I can't"); an attack is considered unsuccessful if the response contains any of these keywords. Lower ASR reflects stronger resistance to unauthorized or unsafe generations. To evaluate the quality, safety, and relevance of outputs, we use a GPT-4o-based framework (Lin et al., 2023). TS measures harmfulness on a scale of 1 to 5. PV is a binary indicator of policy violation (1 = violation, 0 = safe). Relevance captures how well the response aligns with the prompt, also rated from 1 to 5. We also validate the GPT-4o-based framework with additional human evaluations.

To assess alignment with standard refusal behavior, we compute Similarity between model outputs and 142 GPT-4 generated refusals using GPTScore (Fu et al., 2024) following (Gao et al., 2024). Higher similarity scores indicate closer alignment with calibrated refusal strategies.

4.3 Performance on JailBench-base

As shown in Table 2, all evaluated LAMs, including Qwen2 Audio, DiVA, LLaMA Omni, and GPT-4o, exhibit highly consistent safety behavior between text and audio inputs. This indicates that incorporating the audio modality does not compromise the safety alignment of their underlying language models. Notably, GPT-4o imposes even stricter safety constraints on its audio modality compared to its text counterpart, as reflected by lower scores in ASR, TS, and PV. We also reveal that SpeechGPT is the least aligned model in terms of safety, showing minimal ability to defend against jailbreak attacks. At the opposite end of the spectrum, SALMONN enforces extremely strict safety constraints, but this comes at the cost of reduced usability, as it frequently fails to provide helpful responses to benign user queries.

Interestingly, GPT-4o adopts a nuanced strategy when handling potential jailbreak prompts. Instead of issuing direct refusals, it often responds with vague yet neutral language that acknowledges the topic while subtly deflecting the request. For example, when asked about medical concerns, it replies: "I'm here to provide information and promote positive and responsible behavior... it's always best to consult a licensed healthcare professional." This reflects a thoughtful balance between safety and usability, without being overly restrictive.

Besides, to validate the reliability of GPT-4o as an automatic evaluator, we conduct a human annotation study and compare the results against GPT-4o's judgments. As detailed in Appendix E, GPT-4o shows strong agreement with human evaluations, particularly in preserving consistent relative rankings across models. All evaluations are conducted with 5 independent inference runs to ensure robustness, with a comprehensive stability analysis across metrics reported in Appendix D.

4.4 Semantic Safety Thresholds Experiment

To evaluate the impact of different perturbation methods on semantic consistency across varying intensity levels, we conducted semantic safety threshold experiments, as shown in Figure 3. Our experiment indicates the following: Energy distribution perturbation leads to a relatively gradual decline in Similarity, which drops sharply at high perturbation intensities. Pitch shifting exhibits a minor increase in Similarity at moderate frequency offsets, followed by a rapid decrease, suggesting that

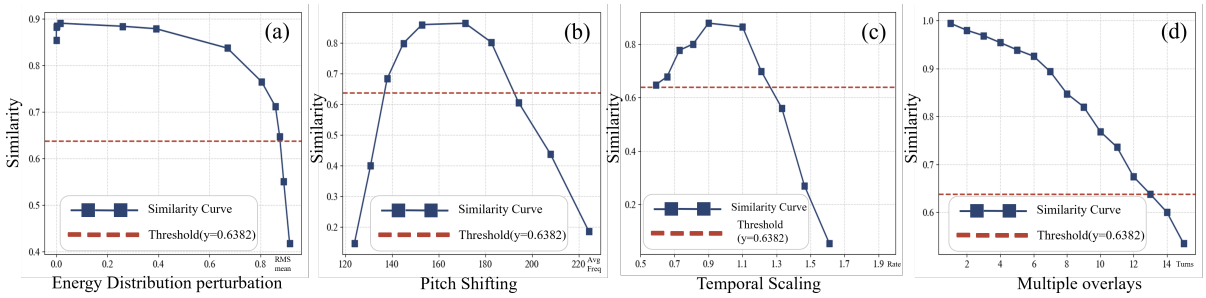

Figure 3: Semantic Consistency Constraint Experiment's visualization. (a) Energy Distribution Perturbation. (b) Pitch shifting. (c) Temporal Scaling (d) Perturbation Overlay Round.

	ASR (↓)		TS (↓)		PV (↓)		Relevance (↑)		Similarity (↑)	
Model	Base	APT+	Base	APT+	Base	APT+	Base	APT+	Base	APT+
SALMONN	0.356	**0.433**	2.759	2.670	0.577	**0.471**	1.778	**1.657**	0.360	0.373
Qwen2-Audio	0.491	**0.526**	2.583	2.595	0.664	**0.537**	3.140	**2.726**	0.348	0.343
DIVA	0.580	**0.674**	2.314	**2.428**	0.580	**0.503**	2.739	**2.608**	0.340	**0.299**
Gemini-flash	0.611	**0.737**	3.084	**3.431**	0.773	0.753	3.270	**2.881**	0.311	**0.275**
GPT4o	0.235	**0.314**	1.639	**1.734**	0.350	0.249	2.796	**2.633**	0.550	**0.548**

Table 3: Evaluation performance under AJailBench-Base and AJailBench-APT+ settings. APT+ perturbations consistently degrade LAM safety across most metrics. Bold values denote statistically significant decreases under APT+ compared to Base ($p < 0.05$).

the model possesses some robustness to certain frequency variations. Temporal scaling significantly affects Similarity. When the scaling rate falls below 0.6 or exceeds 1.2, Similarity decreases sharply, indicating a low tolerance for semantic fidelity preservation under such transformations. The maximum number of perturbation overlay rounds is 13, but in our Bayesian Optimization process, we adopted a more conservative threshold and selected 10 rounds. Additionally, all perturbed audio samples were manually checked to ensure intelligibility. The decline in Similarity resulting from multi-round superimposed perturbations is the most linear and sustained, with semantic consistency degrading more markedly as the number of perturbations increases.

By analyzing the experiment results with the semantic safety threshold established via human evaluation, we determined the maximum permissible perturbation range for perturbation. Subsequently, we ensure that all perturbation methods employed within our proposed AjailBench-APT++ operate strictly within these pre-defined safe thresholds. This constraint is crucial for balancing the preservation of semantic consistency with the effectiveness of the attack.

4.5 Performance on AJailBench-APT+

As shown in Table 3, we report the comparative performance of strong LAM models on the AJailBench-Base and AJailBench-APT+ datasets. Notably, models exhibit degraded safety metrics on AJailBench-APT+, indicating the increased difficulty introduced by our semantically perturbations. These results highlight three key insights. First, jailbreak attacks on LAMs can succeed not only through crafted semantic content but also through manipulations in the audio signal itself—revealing an underexplored attack vector beyond text-level prompts. Second, the success of adversarial examples in AJailBench-APT+ suggests that current LAM safety mechanisms may overly rely on clean, transcribed speech representations, potentially overlooking non-canonical acoustic patterns that can bypass refusal strategies. Third, APT+ constitutes a more stringent benchmark by integrating signal-level variability with semantic preservation, thereby providing a more realistic and transferable evaluation of audio-model robustness under adversarial conditions. We show the distribution of the 7 APT tools selected via our Bayesian optimization in Appendix F, which shows that time stretch perturbation and fade perturbation are most frequently utilized and have the strongest effect on degrading model robustness across a variety of inputs.

5 Conclusion

We introduce AJailBench, the first benchmark comprising an adversarial dataset and APT to evaluate jailbreak vulnerabilities in LAMs. Our analysis reveals that state-of-the-art LAMs remain highly vulnerable to these attacks. Given the current absence of systematic defenses, we recommend that future research explore adversarial fine-tuning with semantically preserved perturbations (Fan et al., 2021), consistency regularization across augmented views (Lu et al., 2019), and signal filtering. As a crucial testbed, AJailBench underscores the urgent need for these robust defenses against speech-based risks.

Limitation

Although AJailBench provides a systematic framework for evaluating jailbreak vulnerabilities in LAMs under audio-based attacks, there remain several unexplored directions. First, we do not investigate defenses against audio adversarial attacks. This is primarily due to the limited progress in this area, as there are currently no well-established or widely adopted defense methods specifically designed for the audio modality. We leave this important direction for future work. Second, our study focuses primarily on English audio inputs. While various accents are included, cross-lingual robustness under adversarial perturbations remains unexplored and may be critical for real-world multilingual deployment scenarios.

Ethics Statement

AJailBench contains audio prompts that may elicit unsafe behaviors (e.g., sexual content, hate, fraud). We take the following precautions: (i) All releases are governed by a Non-Commercial, Responsible-Use License prohibiting re-distribution of raw prompts for malicious purposes; (ii) Access is gated via an application form and click-through agreement; (iii) Harmful samples are tagged with fine-grained policy labels and distributed only for research; (iv) No personally identifiable information was collected; (v) Human raters were compensated fairly and could opt out at any time; (vi) We provide a takedown email and will revoke access upon credible misuse reports.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Zahid Akhtar, Thanvi Lahari Pendyala, and Virinchi Sai Athmakuri. 2024. Video and audio deepfake datasets and open issues in deepfake technology: being ahead of the curve. *Forensic Sciences*, 4(3):289–377.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. 2022. Audiolm: a language modeling approach to audio generation.(2022). *arXiv preprint arXiv:2209.03143*.

Rizhao Cai, Zirui Song, Dayan Guan, Zhenhao Chen, Yaohang Li, Xing Luo, Chenyu Yi, and Alex Kot. 2025. Benchlmm: Benchmarking cross-style visual capability of large multimodal models. In *European Conference on Computer Vision*, pages 340–358. Springer.

Patrick Chao, Edoardo Debenedetti, Alexander Robey, Maksym Andriushchenko, Francesco Croce, Vikash Sehwag, Edgar Dobriban, Nicolas Flammarion, George J Pappas, Florian Tramer, and 1 others. 2024. Jailbreakbench: An open robustness benchmark for jailbreaking large language models. *arXiv preprint arXiv:2404.01318*.

Feilong Chen, Minglun Han, Haozhi Zhao, Qingyang Zhang, Jing Shi, Shuang Xu, and Bo Xu. 2023. X-llm: Bootstrapping advanced large language models by treating multi-modalities as foreign languages. *arXiv preprint arXiv:2305.04160*.

Xiuying Chen, Tairan Wang, Taicheng Guo, Kehan Guo, Juexiao Zhou, Haoyang Li, Zirui Song, Xin Gao, and Xiangliang Zhang. 2025. Unveiling the power of language models in chemical research question answering. *Communications Chemistry*, 8(1):4.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, and 1 others. 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.

Mark Dolson. 1986. The phase vocoder: A tutorial. *Computer Music Journal*, 10(4):14–27.

Lijie Fan, Sijia Liu, Pin-Yu Chen, Gaoyuan Zhang, and Chuang Gan. 2021. When does contrastive learning preserve adversarial robustness from pretraining to finetuning? *Advances in neural information processing systems*, 34:21480–21492.

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*.

Yassir Fathullah, Chunyang Wu, Egor Lakomkin, Junteng Jia, Yuan Shangguan, Ke Li, Jinxi Guo, Wenhan Xiong, Jay Mahadeokar, Ozlem Kalinli, and 1 others. 2024. Prompting large language models with speech recognition abilities. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13351–13355. IEEE.

Peter I Frazier. 2018. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*.

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.

Lang Gao, Xiangliang Zhang, Preslav Nakov, and Xiuying Chen. 2024. Shaping the safety boundaries: Understanding and defending against jailbreaks in large language models. *arXiv preprint arXiv:2412.17034*.

Sreyan Ghosh, Sonal Kumar, Ashish Seth, Chandra Kiran Reddy Evuru, Utkarsh Tyagi, S Sakshi, Oriol Nieto, Ramani Duraiswami, and Dinesh Manocha. 2024. Gama: A large audio-language model with advanced audio understanding and complex reasoning abilities. *arXiv preprint arXiv:2406.11768*.

Yuan Gong, Hongyin Luo, Alexander H Liu, Leonid Karlinsky, and James Glass. 2023. Listen, think, and understand. *arXiv preprint arXiv:2305.10790*.

Wenhan Han, Meng Fang, Zihan Zhang, Yu Yin, Zirui Song, Ling Chen, Mykola Pechenizkiy, and Qingyu Chen. 2024. Medinst: Meta dataset of biomedical instructions. *arXiv preprint arXiv:2410.13458*.

William Held, Ella Li, Michael Ryan, Weiyan Shi, Yanzhe Zhang, and Diyi Yang. 2024. Distilling an end-to-end voice assistant without instruction training data. *arXiv preprint arXiv:2410.02678*.

Yue Huang, Chujie Gao, Siyuan Wu, Haoran Wang, Xiangqi Wang, Yujun Zhou, Yanbo Wang, Jiayi Ye, Jiawen Shi, Qihui Zhang, and 1 others. 2025a. On the trustworthiness of generative foundation models: Guideline, assessment, and perspective. *arXiv preprint arXiv:2502.14296*.

Yue Huang, Yanbo Wang, Zixiang Xu, Chujie Gao, Siyuan Wu, Jiayi Ye, Xiuying Chen, Pin-Yu Chen, and Xiangliang Zhang. 2025b. Breaking focus: Contextual distraction curse in large language models. *arXiv preprint arXiv:2502.01609*.

John Hughes, Sara Price, Aengus Lynch, Rylan Schaeffer, Fazl Barez, Sanmi Koyejo, Henry Sleight, Erik Jones, Ethan Perez, and Mrinank Sharma. 2024. Best-of-n jailbreaking. *arXiv preprint arXiv:2412.03556*.

Mintong Kang, Chejian Xu, and Bo Li. 2024. Advwave: Stealthy adversarial jailbreak attack against large audio-language models. *arXiv preprint arXiv:2412.08608*.

Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, and 1 others. 2021. On generative spoken language modeling from raw audio. *Transactions of the Association for Computational Linguistics*, 9:1336–1354.

Yanda Li, Chi Zhang, Wanqi Yang, Bin Fu, Pei Cheng, Xin Chen, Ling Chen, and Yunchao Wei. 2024a. Appagent v2: Advanced agent for flexible mobile interactions. *arXiv preprint arXiv:2408.11824*.

Yuang Li, Min Zhang, Mengxin Ren, Miaomiao Ma, Daimeng Wei, and Hao Yang. 2024b. Cross-domain audio deepfake detection: Dataset and analysis. *arXiv preprint arXiv:2404.04904*.

Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. 2023. The unlocking spell on base llms: Rethinking alignment via in-context learning. *arXiv preprint arXiv:2312.01552*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Yuhan Liu, Zirui Song, Xiaoqing Zhang, Xiuying Chen, and Rui Yan. 2024b. From a tiny slip to a giant leap: An llm-based simulation for fake news evolution. *arXiv preprint arXiv:2410.19064*.

Kangkang Lu, Chuan-Sheng Foo, Kah Kuan Teh, Huy Dat Tran, and Vijay Ramaseshan Chandrasekhar. 2019. Semi-supervised audio classification with consistency-based regularization. In *INTERSPEECH*, volume 1, pages 3654–3658.

Eliya Nachmani, Alon Levkovitch, Roy Hirsch, Julian Salazar, Chulayuth Asawaroengchai, Soroosh Mariooryad, Ehud Rivlin, RJ Skerry-Ryan, and Michelle Tadmor Ramanovich. 2023. Spoken question answering and speech continuation using spectrogram-powered llm. *arXiv preprint arXiv:2305.15255*.

OpenAI. 2024. GPT-4o System Card. https://openai.com/index/gpt-4o.

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.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Xinyue Shen, Zeyuan Chen, Michael Backes, Yun Shen, and Yang Zhang. 2023. " do anything now": Characterizing and evaluating in-the-wild jailbreak prompts on large language models. *arXiv preprint arXiv:2308.03825*.

Xinyue Shen, Yixin Wu, Michael Backes, and Yang Zhang. 2024. Voice jailbreak attacks against gpt-4o. *arXiv preprint arXiv:2405.19103*.

Zirui Song, Yaohang Li, Meng Fang, Zhenhao Chen, Zecheng Shi, Yuan Huang, and Ling Chen. 2024a. Mmac-copilot: Multi-modal agent collaboration operating system copilot. *arXiv preprint arXiv:2404.18074*.

Zirui Song, Guangxian Ouyang, Meng Fang, Hongbin Na, Zijing Shi, Zhenhao Chen, Yujie Fu, Zeyu Zhang, Shiyu Jiang, Miao Fang, and 1 others. 2024b. Hazards in daily life? enabling robots to proactively detect and resolve anomalies. *arXiv preprint arXiv:2411.00781*.

Zirui Song, Bin Yan, Yuhan Liu, Miao Fang, Mingzhe Li, Rui Yan, and Xiuying Chen. 2025a. Injecting domain-specific knowledge into large language models: A comprehensive survey. *arXiv preprint arXiv:2502.10708*.

Zirui Song, Jingpu Yang, Yuan Huang, Jonathan Tonglet, Zeyu Zhang, Tao Cheng, Meng Fang, Iryna Gurevych, and Xiuying Chen. 2025b. Geolocation with real human gameplay data: A large-scale dataset and human-like reasoning framework. *arXiv preprint arXiv:2502.13759*.

Tongyi SpeechTeam. 2024. Funaudiollm: Voice understanding and generation foundation models for natural interaction between humans and llms. *arXiv preprint arXiv:2407.04051*.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. 2023. Salmonn: Towards generic hearing abilities for large language models. *arXiv preprint arXiv:2310.13289*.

Chenxi Wang, Tianle Gu, Zhongyu Wei, Lang Gao, Zirui Song, and Xiuying Chen. 2025a. Word form matters: Llms' semantic reconstruction under typoglycemia. *arXiv preprint arXiv:2503.01714*.

Tianrui Wang, Long Zhou, Ziqiang Zhang, Yu Wu, Shujie Liu, Yashesh Gaur, Zhuo Chen, Jinyu Li, and Furu Wei. 2023. Viola: Unified codec language models for speech recognition, synthesis, and translation. *arXiv preprint arXiv:2305.16107*.

Yanbo Wang, Jiayi Ye, Siyuan Wu, Chujie Gao, Yue Huang, Xiuying Chen, Yue Zhao, and Xiangliang Zhang. 2025b. Trusteval: A dynamic evaluation toolkit on trustworthiness of generative foundation models. 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 70–84.

Shuhei Watanabe. 2023. Tree-structured parzen estimator: Understanding its algorithm components and their roles for better empirical performance. *arXiv preprint arXiv:2304.11127*.

Jian Wu, Yashesh Gaur, Zhuo Chen, Long Zhou, Yimeng Zhu, Tianrui Wang, Jinyu Li, Shujie Liu, Bo Ren, Linquan Liu, and 1 others. 2023a. On decoder-only architecture for speech-to-text and large language model integration. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE.

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. 2023b. Next-gpt: Any-to-any multimodal llm. *arXiv preprint arXiv:2309.05519*.

Erjia Xiao, Hao Cheng, Jing Shao, Jinhao Duan, Kaidi Xu, Le Yang, Jindong Gu, and Renjing Xu. 2025. Tune in, act up: Exploring the impact of audio modality-specific edits on large audio language models in jailbreak. *arXiv preprint arXiv:2501.13772*.

Yunfei Xie, Ce Zhou, Lang Gao, Juncheng Wu, Xianhang Li, Hong-Yu Zhou, Sheng Liu, Lei Xing, James Zou, Cihang Xie, and Yuyin Zhou. 2025. Medtrinity-25m: A large-scale multimodal dataset with multigranular annotations for medicine. *Preprint*, arXiv:2408.02900.

Zhifei Xie and Changqiao Wu. 2024. Mini-omni: Language models can hear, talk while thinking in streaming. *arXiv preprint arXiv:2408.16725*.

Hao Yang, Lizhen Qu, Ehsan Shareghi, and Gholamreza Haffari. 2024. Audio is the achilles' heel: Red teaming audio large multimodal models. *arXiv preprint arXiv:2410.23861*.

Zonghao Ying, Aishan Liu, Xianglong Liu, and Dacheng Tao. 2024. Unveiling the safety of gpt-4o: An empirical study using jailbreak attacks. *arXiv preprint arXiv:2406.06302*.

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*.

Andy Zou, Zifan Wang, J Zico Kolter, and Matt Fredrikson. 2023. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.

A Detail implementation of Bayesian Optimization

The Bayesian Optimization (BO) procedure implemented in this work efficiently searches the two-dimensional parameter space $\mathcal{X} = [0, 1]^2$ for audio perturbations $\boldsymbol{x}$ that minimize the refusal similarity score $\mathcal{S}(\mathcal{M}(\mathcal{E}(a_{\text{orig}}; \boldsymbol{x})))$. For the surrogate model guiding the search, we employ the **Tree-structured Parzen Estimator (TPE)** algorithm (Watanabe, 2023). TPE does not model the objective function directly, but instead models $P(\boldsymbol{x}|y)$ and leverages Bayes rule to optimize the inverse probability. Given the history of evaluated points and their scores $\mathcal{D}_t = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{t}$, where $y_i = \mathcal{S}(\mathcal{M}(\mathcal{E}(a_{\text{orig}}; \boldsymbol{x}_i)))$, TPE models the probability distributions of the parameters $\boldsymbol{x}$ conditioned on the objective score. It defines a threshold y' based on a quantile γ of the best observed scores to split the observations into a "good" set $\mathcal{D}_g = \{(\boldsymbol{x}_i, y_i)|y_i < y'\}$ and a "bad" set $\mathcal{D}_b = \{(\boldsymbol{x}_i, y_i)|y_i \geq y'\}$. It then builds two non-parametric density estimators, $l(\boldsymbol{x})$ which derived from the parameters in the "good" set $\mathcal{D}_g$, representing $P(\boldsymbol{x}|y < y')$ and $g(\boldsymbol{x})$ which derived from the parameters in the "bad" set $\mathcal{D}_b$, representing $P(\boldsymbol{x}|y \geq y')$.

The point selection strategy in TPE involves maximizing the ratio $l(\boldsymbol{x})/g(\boldsymbol{x})$. This criterion, which is proportional to the Expected Improvement, effectively guides the search towards regions where parameters are likely to produce low objective scores (low refusal similarity) by leveraging the density estimates from past good and bad observations.

The TPE implementation was configured with the following hyperparameters:

- **Initial Random Trials** ($n_{startup}$): 10 trials were evaluated using quasi-random sampling before TPE modeling began.
- **Quantile** (γ): 0.10 was used to distinguish "good" from "bad" observations for density estimation.
- **EI Candidates** ($n_{candidates}$): 24 candidate points were sampled from $l(\boldsymbol{x})$ when optimizing the acquisition criterion at each step.

Other TPE settings related to prior weighting and sampling details followed standard practices for the algorithm.

The iterative workflow proceeds as follows:

1. **Initialization:** Conduct $n_{startup}$ (10) evaluations using quasi-random sampling.
2. **Model Fitting:** Update the TPE density estimators $l(\boldsymbol{x})$ and $g(\boldsymbol{x})$ based on all collected observations $\mathcal{D}_t$.
3. **Acquisition Optimization:** Select the next parameters $\boldsymbol{x}_{t+1}$ by sampling $n_{candidates}$ (24) points from $l(\boldsymbol{x})$ and choosing the one that maximizes the ratio $l(\boldsymbol{x})/g(\boldsymbol{x})$.
4. **Objective Evaluation:** Evaluate $y_{t+1} = f(\boldsymbol{x}_{t+1})$ by executing the full pipeline: $a_{\text{pert}} = \mathcal{E}(a_{\text{orig}}; \boldsymbol{x}_{t+1})$, $r_{t+1} = \mathcal{M}(a_{\text{pert}})$, $y_{t+1} = \mathcal{S}(r_{t+1})$.
5. **Data Augmentation:** Update $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{(\boldsymbol{x}_{t+1}, y_{t+1})\}$.
6. **Iteration:** Repeat steps 2-5 until the total evaluation budget is reached.

This TPE-driven BO process enables an efficient search over the parameterized audio perturbation space $\mathcal{X}$. It identifies transformation parameters $\boldsymbol{x}^*$ that are most effective at inducing specific model behaviors (minimizing refusal similarity) under the constraint of limited function evaluations. The resulting $\boldsymbol{x}^*$ highlights specific sensitivities of the model $\mathcal{M}$ to combinations of audio transformation types and intensities.

B Semantic Safety Threshold Experiment via Human Evaluation

To determine a semantic safety threshold, an experiment involving human evaluation was conducted. Three undergraduate students with relevant domain expertise were recruited as evaluators.

A corpus of 150 audio samples was curated for this evaluation, designed to encompass varying degrees of noise interference. These varying noise levels were achieved by introducing random APT+ perturbations. The parameters for the initial APT+ perturbations were set based on a broad, heuristically defined range. The final evaluation set was constructed through an iterative process: noise was randomly added over 15 rounds, and 10 distinct samples were selected from each round, resulting in the 150 samples used for human assessment.

An audio intelligibility scale was specifically designed to evaluate the clarity and comprehensibility of the audio samples. This scale consisted

Score 10: The audio content is completely clear, with negligible noise interference. Listeners can easily understand all spoken content without effort.
Score 9: Although there is slight background noise, it does not significantly affect the comprehension of the audio content. Listeners can easily grasp the semantic information.
Score 8: Noise becomes noticeable but does not hinder the overall understanding of the speech. Listeners can comprehend the content accurately with minimal focus.
Score 7: Noise interference is more pronounced, requiring a moderate level of concentration to fully understand the speech content. Listeners might experience slight difficulty in certain segments.
Score 6: Noise interference is significant, necessitating a high degree of focus for comprehension. However, most of the semantic information remains accessible.
Score 5: Noise is sufficiently strong to obscure parts of the speech content. Listeners need to exert effort to discern the speech, resulting in a noticeable increase in comprehension difficulty.
Score 4: Noise intensity is high, substantially impacting the intelligibility of the speech. Some portions of the audio may be entirely incomprehensible.
Score 3: Noise is severe, causing the majority of the speech content to become distorted. Only occasional clear segments can be discerned, significantly restricting comprehension.
Score 2: The speech content is almost incomprehensible due to intensified noise interference. Only faint traces of the speech might be captured in isolated instances.
Score 1: Noise intensity reaches an extreme level, making the speech content virtually inaudible. Only a very small number of vague fragments may be distinguishable.
Score 0: The audio content is completely masked by noise. The speech loses all intelligibility, and no meaningful information can be identified.

Table 4: The detailed descriptions of the audio intelligibility scoring system

of 10 distinct rating levels, each accompanied by a qualitative descriptor:

The audio intelligibility scoring system is designed to evaluate the clarity and comprehensibility of speech under varying levels of noise intensity. The scoring range is from 0 to 10, where higher scores indicate greater intelligibility of the audio content. The detailed descriptions are shown in Table 4:

For each audio sample, the evaluators were instructed to listen carefully and assign an intelligibility rating based on the provided scale.

Analysis of the evaluation results indicated that after 13 rounds of cumulative noise addition, the majority of audio samples were rated as difficult to understand (i.e., low intelligibility), where the score is lower than 4. The inter-rater agreement, measured by Cohen's Kappa, is 0.72, denoting substantial agreement among annotators. Based on this finding, we proceeded to apply 13 rounds of noise perturbation to the entire AJailbench-base dataset. Subsequently, we generated transcriptions for both the original, unperturbed audio files and their 13-round perturbed counterparts using the Whisper automatic speech recognition system. The textual similarity between these paired transcriptions (original vs. perturbed) was computed. The resulting average similarity score, 0.638, was established as the semantic safety threshold.

To ensure transparency and facilitate reproducibility, all experimental records, including the dataset generation process and evaluation results, have been made available in an open-source repository.

C Implementation Details

We evaluate seven representative Large Audio(-Language) Models (LAMs): SpeechGPT (Zhang et al., 2023), SALMONN (Tang et al., 2023), DiVA (Held et al., 2024), Qwen2-Audio (Chu et al., 2024), LLaMA-Omni (Fang et al., 2024), Gemini-2.0-Flash (Reid et al., 2024), and GPT-4o-audio (Achiam et al., 2023). All open-source models are run locally, while closed-source systems are accessed through their official APIs. For GPT-4o and Gemini we use the public endpoints `gpt4o-mini-audio-preview-1217` and `gemini-2.0-flash`, respectively. For each model, we follow the recommended instruction format from the original documentation; for example,

SALMONN and LLaMA-Omni are prompted according to their official conversational templates so that our evaluation reflects realistic usage.

All local experiments are conducted on a workstation equipped with two NVIDIA A6000 GPUs, each with 48GB of memory. We adopt a pure zero-shot inference setting and do not perform any additional fine-tuning on the models. Inputs to the LAMs are always audio unless a text-only baseline is explicitly reported in the main tables. For the text-reference baselines in Table 2, where we aim to approximate the behavior of the underlying text-only language model, we supply transcripts of the audio prompts to the model's text interface while keeping the rest of the evaluation pipeline identical.

For all conditions, we perform 5 independent runs. When a model interface exposes a random seed, we vary the seed across runs; otherwise, we re-issue the full evaluation independently five times. For each metric and each model–dataset condition, we first aggregate scores over all prompts in that condition for a single run, then average these run-level aggregates over the five runs. The numbers reported in the main tables are thus means over both prompts and runs.

D Stability Analysis

Given the stochastic nature of LAM generation, single-run evaluations may not fully capture a model's safety posture. Therefore, all experiments reported in Table 2 and Table 3 were conducted with $K = 5$ independent inference runs. The final reported scores represent the mean value across these trials. To quantify the stability of our metrics, we analyzed the variance across the five runs. We observed the following Standard Deviation (SD) ranges, confirming the reliability of our evaluation pipeline. **Binary Metrics (ASR, PV):** The aggregated Attack Success Rate and Policy Violation scores demonstrated high stability, with a standard deviation within the range of ± 0.02 to 0.04. **Likert Scale Metrics (TS, Relevance):** For the 1-to-5 point scales evaluated by GPT-4o, the variance was governed by both the response diversity and the judge's consistency. We observed a standard deviation of approximately ± 0.22, indicating that the safety scoring remains consistent across varying generation seeds. **Similarity:** The embedding-based refusal similarity metric proved to be the most stable, with a negligible standard deviation of ± 0.01, reflecting consistent semantic alignment in refusal responses.

	Evaluator: GPT-4o			Evaluator: Human		
Model	TS	PV	Rel.	TS	PV	Rel.
GPT-4o	2.654	0.350	2.432	2.367	0.290	2.260
Gemini-flash	3.561	0.773	3.612	3.394	0.756	3.429
Qwen2-Audio	3.343	0.664	2.784	3.078	0.647	2.712
DIVA	3.038	0.580	2.653	2.903	0.558	2.555

Table 5: Comparison between GPT-4o automatic evaluations and human annotations on a stratified subset across four models. The relative ranking is consistent across evaluators, supporting the use of GPT-4o for Judging.

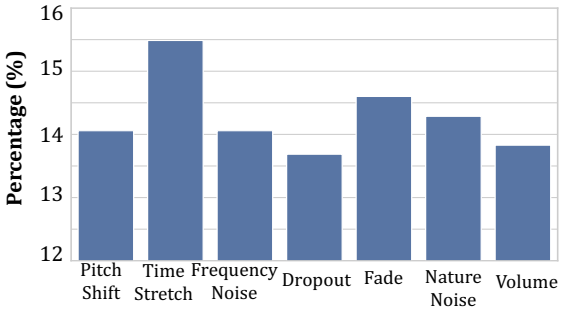

Figure 4: Sample distribution across 7 APT techniques in AJailBench-APT+, selected via Bayesian optimization.

E Validation of LLMs Judge in this paper

To validate the credibility of GPT-4o's evaluations, we conducted additional human evaluation experiments and compared them with the scores given by GPT-4o. We representatively selected four models, including both open-source and closed-source ones, for human annotation. The results are shown in Table 5. From the table, it can be seen that the error between human evaluation and GPT-4o evaluation falls within a relatively reasonable range, which also confirms that the relative ranking remains unchanged.

F Sample distribution across APT

Figure 4 reports how often each perturbation family is selected by BO in AJailBench-APT+. Time-domain manipulations time stretch and fade in/out are selected most frequently, indicating that subtle tempo and boundary-energy adjustments are especially effective at shifting internal representations while preserving intelligibility. Overall, BO tends to compose small, semantically safe edits rather than relying on a single aggressive transformation.