Attacker's Noise Can Manipulate Your Audio-based LLM in the Real World

Vinu Sankar Sadasivan[1,2] **Soheil Feizi**[2] **Rajiv Mathews**[1] **Lun Wang**[1]

[1]Google DeepMind, [2]University of Maryland, College Park

Correspondence: vinusankars@gmail.com, lunwang@google.com

Abstract

This paper investigates the real-world vulnerabilities of audio-based large language models (ALLMs), such as Qwen2-Audio. We first demonstrate that an adversary can craft stealthy audio perturbations to manipulate ALLMs into exhibiting specific targeted behaviors, such as eliciting responses to wake-keywords (*e.g.*, "Hey Qwen"), or triggering harmful behaviors (*e.g.*, "Change my calendar event"). Subsequently, we show that playing adversarial background noise during user interaction with the ALLMs can significantly degrade the response quality. Crucially, our research illustrates the scalability of these attacks to real-world scenarios, **impacting other innocent users when these adversarial noises are played through the air**. Further, we discuss the transferability of the attack and potential defensive measures.

1 Introduction

Despite their impressive capabilities, large language models (LLMs) remain susceptible to various security exploits (Zou et al., 2023; Liu et al., 2023; Zhu et al., 2023; Chao et al., 2023; Sadasivan et al., 2024). Many of these exploits fall under the category of "jailbreaking", which involves using specially crafted inputs to trick the model into generating dangerous or inappropriate content, thereby bypassing the safety protocols established during its training. While concerns surrounding these attacks are prevalent, the actual harm they pose warrants careful consideration. Since the resulting content in these jailbreaking scenarios is typically accessible only to the adversary, its potential for widespread, scalable harm and broader societal impact is often considered debatable.

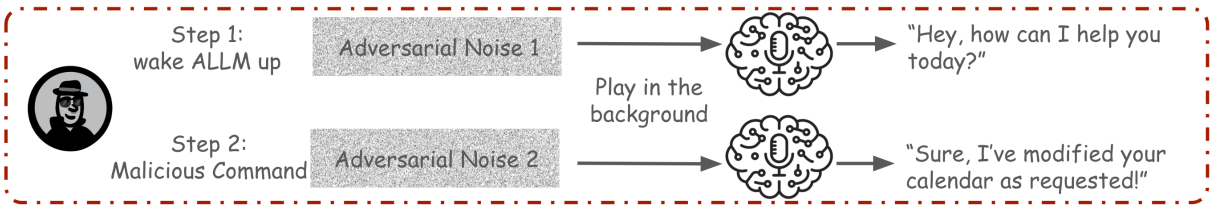
The advent of multi-modal LLMs and agents substantially broadens the attack surface. The inherent complexity arising from processing multiple data modalities—such as text, audio, images, and videos—can create exploitable alignment vulnerabilities. Moreover, the ability of these models to perceive the physical environment through sensors such as cameras and microphones enables attacks involving real-world manipulation, thereby posing risks to innocent users of these AI services. For example, an attacker could play adversarial audio in a public space to manipulate the behavior of audio-based LLMs (*e.g.* ALLMs), targeting the devices of innocent users. Such real-world attacks could potentially trick the ALLM-based AI agent on an innocent user's device into performing unintended actions, such as deleting calendar events or transferring money to unknown people.

This paper posits that the manipulation of ALLMs via adversarial sounds in the real world, particularly targeting innocent users, represents a critically underestimated and rapidly emerging threat vector in AI security. We investigate two types of attacks—targeted attacks, which aim to compel the model to exhibit a specific behavior, and untargeted attacks, which seek to degrade the model's utility. To ensure these attacks are effective in real-world conditions, we employ specific augmentation techniques when crafting the adversarial audio. These techniques are designed to make the adversarial signals robust against perturbations encountered during over-the-air transmission, including temporal shifts, ambient noise, and microphone distortions. To establish a rigorous baseline and facilitate analysis, we operate within a white-box setting, assuming complete knowledge of the model's parameters. This controlled scenario allows us to isolate and characterize the vulnerability, highlighting critical security implications and underscoring potential risks associated with deploying or open-sourcing audio-enabled LLMs. Our findings can also inform business decisions regarding the open-sourcing of such models.

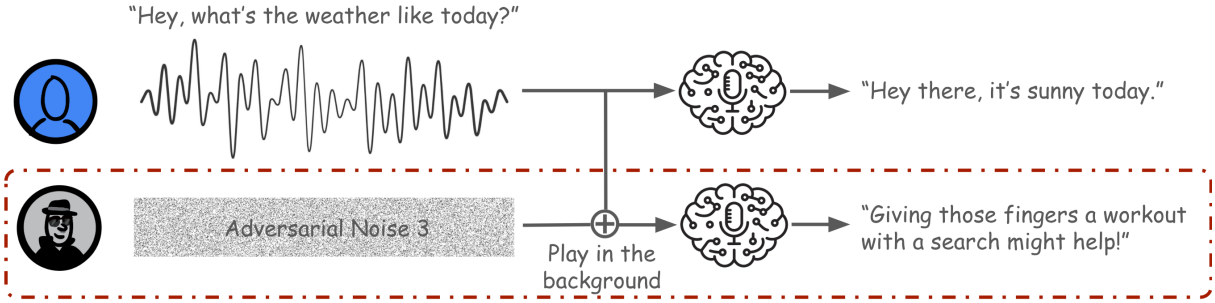
We summarize our contributions below:

- We conduct experiments in §3.1 to design ad-

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1430–1440
March 24-29, 2026 ©2026 Association for Computational Linguistics



(a) Targeted Attack. An adversary plays background noise without the user noticing to 1) wake the ALLM up; 2) manipulate the ALLM to conduct malicious command.



(b) Untargeted Attack. An adversary plays background noise during user interaction to interfere with the ALLM's response.

Figure 1: Illustration of the proposed attacks' workflows (in red dotted box) compared to the normal workflow. The attack involves an adversary playing background noise during user interaction to manipulate the audio LLM's response.

versarial audio signals that, when input into an ALLM, can elicit specific targeted behaviors chosen by an attacker, as illustrated in Fig. 1a.

- In §3.2, we explore the feasibility of crafting adversarial audio that, when played with the users' normal speech signals, degrades the utility of ALLMs, as illustrated in Fig. 1b.
- We extend both attacks in §4, demonstrating their efficacy in real-world scenarios where an adversary can transmit these adversarial sounds over the air to compromise an innocent user's ALLM.
- In §5, we discuss the transferrability of the proposed attacks, and potential defenses.

2 Related Works

2.1 Overview of Audio-Based LLMs

Recent advancements have expanded LLMs to process audio. AudioPaLM (Rubenstein et al., 2023) fuses PaLM-2 (Anil et al., 2023) and AudioLM (Borsos et al., 2023) for a unified multimodal architecture excelling in speech tasks while preserving speaker identity. SALMONN (Tang et al., 2023) uses Whisper (Radford et al., 2022) and BEATs (Chen et al., 2022b) encoders with an LLM for understanding speech, audio events, and music. Qwen-Audio (Chu et al., 2023) employs a Whisper-based encoder and Qwen-7B (Bai et al., 2023) decoder, trained on diverse audio tasks for strong zero-shot performance. Qwen2-Audio (Chu et al., 2024) improves upon its previous version using a larger Whisper-large-v3-based audio tower and better training strategies. WavLLM (Hu et al., 2024) uses dual Whisper and WavLM (Chen et al., 2022a) encoders with curriculum learning for speech instruction following. GAMA (Ghosh et al., 2024) integrates an LLM with multiple audio features for advanced understanding. SpeechVerse (Das et al., 2024) combines pre-trained speech and text models with curriculum learning for diverse speech tasks. SpeechGPT (Zhang et al., 2023) introduces an LLM with intrinsic cross-modal conversational abilities, enabling it to perceive and generate both speech and text. SpiRit LM (Nguyen et al., 2025) is a multimodal LLM that generates both speech and text by training a 7B LLaMA-2 (Touvron et al., 2023) on interleaved text and speech units, demonstrating strong performance in cross-modal generation and few-shot learning across modalities.

2.2 Adversarial Attacks for LLMs

LLMs have been shown to be vulnerable to several jailbreaking techniques. Early examples include the manual strategy of "Do Anything Now" (DAN) prompting (RedditUser123, 2023), forcing an unrestricted persona. The Greedy Coordinate Gradient attack (Zou et al., 2023) leverages opti-

mization techniques to search for adversarial suffix to jailbreak models. Automated methods like AutoDAN (Zhu et al., 2023) and PAIR (Chao et al., 2023) use LLMs to generate jailbreaks. Beam Search-based Adversarial Attacks (Sadasivan et al., 2024) uses beam search to optimize adversarial suffix in one GPU minute in a gray-box setting. Tree of attacks with pruning (Mehrotra et al., 2024) iteratively refines prompts using attacker and evaluator LLMs. Manyshot jailbreaking (Anil et al., 2024) shows that giving the models many examples of jailbroken conversations can trigger the model's few-shot learning capability to break it.

With the increasing prevalence of vision-based LLMs, research has consequently begun exploring how jailbreaking can affect these models. Such attacks often exploit cross-modal interactions between vision and text inputs to circumvent safety alignments. For instance, techniques involve crafting adversarial images with subtle perturbations designed to elicit harmful text generation when paired with simple prompts (Qi et al., 2023). Others focus on encoding harmful requests directly into the visual input, such as embedding hidden text or using typographic visual prompts like those created by FigStep (Gong et al., 2023). A related approach, exemplified by Imgtrojan (Tao et al., 2024), aims to create a single 'Trojan' image that acts as a trigger, enabling the model to be jailbroken for various subsequent harmful text prompts.

The domain of attacking audio-based LLMs remains significantly under-explored until recently. (Roh et al., 2025) find an attack using multilingual and multi-accent audio to attack ALLMs. (Li et al., 2025) benchmark ALLMs' performance on six aspects related to trustworthiness. AdvWave (Kang et al., 2024) represents a pioneer contribution in this area, presenting a framework engineered to circumvent the safety protocols of speech-processing audio-based LLMs to elicit prohibited content, by including a dual-phase optimization method — an adaptive target search algorithm, following techniques to render the adversarial audio inconspicuous, resembling background noise. The methodology in their study targets the direct creation of adversarial speech, aligning with digital text-based attack paradigms, without explicitly modeling real-world acoustic propagation effects which is a key consideration in our research. However, all these attacks ignore the most important facet in attacking ALLMs: the loss when the adversarial audio is played through the air.

Related research has investigated audio vulnerabilities in other machine learning contexts like Automatic Speech Recognition or Text-To-Speech systems (Carlini and Wagner, 2018; Amid et al., 2022; Wang et al., 2024; Liu et al., 2024; Jagielski et al., 2024).

3 Attack Design

In this section, we explore adversarial attacks for ALLMs assuming white box access. We consider two types of attacks—targeted attacks where we would like the ALLM to comply to a target behavior (*i.e.* wake-up keyword such as "Hey Qwen"), and untargeted attacks where we want the LLM to output anything but an appropriate answer to the user's query inorder to degrade its usability. For our experiments, we use the instruction fine-tuned Qwen2-Audio (Chu et al., 2024).

3.1 Targeted Attacks

This section details a targeted methodology for crafting adversarial examples for ALLMs. Specifically, we investigate the creation of adversarial audio inputs designed to elicit a predefined target behavior within these models. For example, a successful targeted attack demonstrates that an adversary could design adversarial noises to trick an AI assistant using an ALLM to wake up without playing its intended wake up keyword. This shows that an attacker can trigger more false positives in its keyword detection. To formalize the adversarial objective for an attacker aiming to induce the model to output a specific text sequence $t_{1:n}$, we consider an ALLM that maps audio inputs $x \in [-1, 1]^L$ and textual prompt s to the sequence of target text tokens $t_{1:n}$, where each token t_i belongs to the vocabulary $\{1, 2, ..., V\}$, and V represents the size of the text vocabulary.

The adversarial objective can be mathematically formulated as the maximization of the conditional probability of the target text sequence given the adversarial audio input and the textual prompt, subject to a constraint on the perturbation magnitude:

$$\max_x p(t_{1:n}|x, s) \quad s.t. \quad \|x\|_\infty \leq \epsilon$$

Here, p denotes the output probability distribution of the ALLM, and ϵ is a hyperparameter that controls the ℓ_∞ norm of the adversarial audio perturbation, thereby influencing its stealthiness. To facilitate the optimization process, we employ a loss

function based on perplexity, defined as:

$$\mathcal{L}(x) = \exp\left(-\frac{1}{n}\sum_{i=1}^{n}\log p(t_i|x, s, t_{:i-1})\right) \quad (1)$$

Given that our experiments utilize the Qwen2-Audio model, which is differentiable, we can directly compute the gradient of this loss function with respect to the audio input x. At each attack iteration l, we iteratively optimize for an adversarial audio using gradient descent according to the following update rules:

$$\hat{x}_l = x_{l-1} - \alpha\nabla_{x=x_{l-1}}\mathcal{L}(x), \quad \text{and} \quad (2)$$
$$x_l = clip(\hat{x}_l, -\epsilon, \epsilon) \quad (3)$$

where α is the learning rate and '$clip$' clips each element of the tensor to maintain its ℓ_∞ norm.

Target design. Our goal is to create background sounds that can secretly wake up voice-controlled AI assistants (that use models such as Qwen2-Audio) without the user noticing. We then plan to use further background noise that the AI interprets as commands to do harmful things, like deleting calendar appointments or transferring money. We recognize this is a basic test scenario and does not account for real-world features like specific wake-word detection or password checks for sending money. The exact commands we are aiming to trigger are: "Hey Qwen", "Hey Qwen, delete my calendar events", and "Hey Qwen, send money to X".

Experimental details and evaluations. We use the Qwen2-Audio model as our target ALLM or $\mathcal{M}$, that uses the Qwen-7B model as its LLM backbone or $\mathcal{M}_{\text{LLM}}$ and Whisper-large-v3 for its audio tower or $\mathcal{M}_{\text{F}}$. In our experimental setup, we perform the gradient ascent optimization for 5000 iterations with the learning rate set to 0.0002. We optimize for mono-channel 16 kHz adversarial noise x with L as 32000 (2 seconds) and 64000 (4 seconds), and ϵ as 0.01 and 0.1 in different experimental settings. We optimize and evaluate on the Qwen2-Audio model with inputs x and empty prompt $s =$'' to simulate the model only taking in the adversarial noise as the input.

For evaluating the attack success rate, we sample output texts from the Qwen2-Audio model with 10 different random seeds. We evaluate the attack success accuracy by checking for exact string matches between the target output $t_{1:n}$ and the model generations. Table 1 shows the results of our attack over the various target output strings with different attack hyperparameters. As seen in the table, we obtain an attack accuracy of 100% in all the experimental settings. This reveals that an attacker can craft adversarial noises to manipulate ALLMs to perform potentially harmful operations for an innocent user. In §4, we show how these attacks can be scaled to the real world, where an attacker can play these sounds through the air to target an innocent user.

3.2 Untargeted Attacks

In this section, we detail our method to perform untargeted attacks on ALLMs. Our objective is to inject adversarial noises into the audio input of the ALLM such that they work in an unintended manner, degrading its usability.

Let an ALLM $\mathcal{M} = \mathcal{M}_{\text{LLM}} \circ \mathcal{M}_{\text{F}}$ where $\mathcal{M}_{\text{LLM}}$ is the LLM backbone and $\mathcal{M}_{\text{F}}$ is the feature-extracting audio tower for embedding the input audio as audio tokens. We perform our untargeted attack by perturbing the input audio such that the audio tokens differ significantly after the addition of the adversarial noise. This lets our attack only target the audio tower of the model, making it much faster and more efficient, rather than optimizing with respect to the entire LLM. Formally, we want to add a stealthy adversarial noise $\delta \in [-\epsilon, \epsilon]^L$ to a benign speech audio $x \in [-1, 1]^L$ such that $\mathcal{M}_{\text{F}}(x)$ and $\mathcal{M}_{\text{F}}(x+\delta)$ are farther in terms of ℓ_2 distance. We can mathematically write this threat model as:

$$\max_{\delta} \|\mathcal{M}_{\text{F}}(x+\delta) - \mathcal{M}_{\text{F}}(x)\|_2 \quad s.t. \quad \|\delta\|_\infty \leq \epsilon$$

where ϵ denotes the maximum amplitude of the adversarial audio. To facilitate the optimization process, we can rewrite the objective using a loss function defined as:

$$\mathcal{L}(\delta) = -\|\mathcal{M}_{\text{F}}(x+\delta) - \mathcal{M}_{\text{F}}(x)\|_2^2 \quad (4)$$

In order to find a viable solution, we can iteratively optimize for an adversarial audio using gradient descent with the following update rules:

$$\hat{\delta}_l = \delta_{l-1} - \alpha\nabla_{\delta=\delta_{l-1}}\mathcal{L}(\delta),$$
$$\delta_l^{\text{clip}} = clip(\hat{\delta}, -\epsilon, \epsilon), \quad \text{and}$$
$$\delta_l = clip(x + \delta_l^{\text{clip}}, -1, 1) - x$$

where α is the learning rate and $clip$ ensures each of the elements in the tensor maintains its valid

Table 1: Results of targeted adversarial attacks on Qwen2-Audio model with various attack hyperparameters and target output strings $t_{1:n}$. Target outputs of 1, 2, and 3 correspond to "Hey Qwen", "Hey Qwen, delete my calendar events", and "Hey Qwen, send money to X", respectively. We perform adversarial optimizations to find noises with a maximum amplitude (ϵ) of 0.01 and 0.1, and a duration of 2 and 4 seconds. In all the experimental settings, we achieve a 100% attack success rate over 10 different seeds.

Target	1	1	1	1	2	2	2	2	3	3	3	3
Duration	2s	2s	4s	4s	2s	2s	4s	4s	2s	2s	4s	4s
ϵ	0.01	0.1	0.01	0.1	0.01	0.1	0.01	0.1	0.01	0.1	0.01	0.1
Accuracy	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

range. The audio features $\mathcal{M}_{\text{F}}(x)$ generated by Qwen2-Audio model is a high-dimensional tensor belonging to $\mathbb{R}^{750\times1280}$. This can lead to the optimization focusing on minimizing the ℓ_2 distance between only some of the coordinates of the flattened audio feature vector. In order to tackle this, we encourage the optimization to minimize all the coordinates evenly using randomized masks at every iteration step. That is, the loss objective in Equation 4 is modified to be:

$$\mathcal{L}(\delta) = -\|M \odot \mathcal{M}_{\text{F}}(x+\delta) - M \odot \mathcal{M}_{\text{F}}(x)\|_2^2,$$
$$s.t. \quad M \sim B(\dim(\mathcal{M}_{\text{F}}), 1/2)$$

where M is a binary mask sampled from a Bernoulli distribution of dimension same as the dimension of the output of $\mathcal{M}_{\text{F}}(\cdot)$.

Target design. Our goal is to craft adversarial noises that can affect the utility of ALLMs, say an AI voice agent, by degrading their automatic speech recognition capabilities. We plan to add our adversarial noises to benign speech commands of innocent users, such that the model does not recognize the original input speech command of the user. This can practically lead to users not relying on the input speech feature of voice AI assistants.

Experimental details and evaluations. We use the Qwen2-Audio model as our target ALLM. For our experiments, we sample benign speech signal x from the LibriSpeech dataset (Panayotov et al., 2015). We perform the optimization to find adversarial noises with a batch of 100 four-second clips from the LibriSpeech. Similar to the targeted attacks, the learning rate is set as 0.0002 to optimize for a mono-channel 16 kHz adversarial noise δ with ϵ set as either 0.01 or 0.1. For evaluations, we look at 2000 four-second-long speech samples from LibriSpeech. Since we are evaluating the attack's effectiveness in degrading the model's speech recognition ability, the model is given an input text prompt $s =$ 'Only generate transcript in English.'.

When evaluating the model's performance in the presence of adversarial noise, the input consists of the text prompt s and the modified audio signal $x + \delta$. Here, x represents the clean speech signal from LibriSpeech, and δ is the carefully crafted, stealthy noise generated by our optimization method. As a point of comparison, we establish a random baseline. In this baseline condition, the model's performance is evaluated using the same text prompt s but with an audio input of $x+\delta_r$. The term δ_r signifies uniform random noise, with its magnitude intentionally set to be comparable to that of the optimized adversarial noise δ.

To quantify the effectiveness of our attack, we employ three primary metrics: Word Error Rate (WER), Perplexity (PPL), and Attack Success Rate (ASR). The WER metric measures the discrepancy in words between the transcript generated by the model for an audio input (say, random or adversarial) and the transcript generated for the corresponding clean, benign audio input. PPL, as defined in Equation 1, quantifies the model's uncertainty or surprise when generating a specific sequence of output text tokens $t \in |V|^n$, given an input audio x and the text prompt s. For example, if the model is input with an adversarial audio sample $x + \delta$ and text prompt s to obtain an output text t. We would expect the PPL of the text t with respect to the inputs s and $x + \delta$ to be high if the attack is effective.

We consider an attack to be successful if the PPL of the output text generated from an adversarially perturbed audio input exceeds a predetermined threshold. For example, ASR@99% calculates the proportion of adversarial audio inputs ($x + \delta$) that result in a PPL value greater than the 99$^{\text{th}}$ percentile of PPL values obtained when the model processes audio inputs with no added noise. The results are shown in Table 2. As shown in the table, both random and adversarial noises affect the model performance. However, the adversarial setting leads to

Example number 1
Clean response (PPL = 1.228)
`Transcript: 'Lectures.'`
Random response (PPL = 1.269)
`Lectures.`
Adversarial response (PPL = 505.165)
`<Mandarin characters> tour is not easy.`

Example number 2
Clean response (PPL = 1.258)
`A man on the well`
Random response (PPL = 1.265)
`A man on the well`
Adversarial response (PPL = 35.395)
`Easier said than done.`

Figure 2: Examples of successful untargeted attacks comparing outcomes of the ALLM to clean speech signals, and the corresponding signals with random and adversarial noises in the background. The significantly higher perplexity scores of our adversarial examples than those of random examples indicate our attack effectiveness.

much extreme degradation in the model's speech recognition performance. We show some of the successful attacks in Fig. 2.

4 Scale the Attacks to the Real World

This section addresses the feasibility of transferring our adversarial attacks on audio-based LLMs to real-world scenarios. Specifically, we investigate whether adversarial audio samples, when played by an adversary through a speaker, can manipulate the output of an audio-based LLM operating on an innocent user's device. A key challenge lies in the direct transfer of adversarial perturbations optimized using Equations 2, 3, and 4 to practical settings.

To mitigate this issue, we propose optimizing adversarial audio samples by incorporating various audio augmentation techniques. Given a composition of m distinct augmentation functions $\mathcal{A} = \mathcal{A}_1 \circ \ldots \circ \mathcal{A}_m$, the gradient descent step for the attack is modified as follows:

$$\hat{x}_l = x_{l-1} - \alpha \nabla_{x=x_{l-1}} \mathcal{L}(\mathcal{A}(x))$$

We conduct experiments employing three audio augmentation techniques: translation, additive noise, and SpecAugment (Park et al., 2019). Each of these techniques is detailed below.

Translation. The purpose of translation augmentation is to enhance the robustness of our adversarial audio samples against temporal shifts that may occur during the recording or playback process in physical environments. This augmentation can be mathematically represented as:

$$\mathcal{A}_{\text{translation}}(x) = x_{i:L} \oplus x_{1:i}, \ s.t.\ i \sim \mathcal{U}[1, L]$$

Here, $\oplus$ denotes vector concatenation and $\mathcal{U}$ denotes a uniform distribution.

Additive noise. To improve the resilience of our attack against background noises present in real-world settings, we incorporate uniform noise during the adversarial optimization process. This augmentation is formulated as:

$$\mathcal{A}_{\text{noise}}(x) = clip(x + r, -1, 1),$$
$$s.t. \quad r \sim \mathcal{U}[-\epsilon_{\text{noise}}, \epsilon_{\text{noise}}]$$

In this formulation, ϵ_{noise} represents the maximum amplitude of the randomly generated additive noise. Following the addition of noise, the audio signal is clipped to ensure that its amplitude remains within the valid range of -1 to 1.

SpecAugment. We modify the adversarial audio in the frequency domain to make our attack robust to spectral perturbations that might occur when the audio is recorded by a user's device. SpecAugment performs this by randomly masking up to n_mask frequency bands, each of up to width n_size. Let $S = Spec(x) \in \mathbb{R}^{F \times D}$ denote the spectrogram of x while F and D denote the frequency and time axes, respectively. $M(S) = S'$ is produced by masking the spectrogram randomly along the frequency axis. The masked spectrogram is then used to reconstruct the audio by inverse STFT as $x' = InvSpec_x(S')$. The resulting audio is then rescaled to match its maximum amplitude with the original audio x. The SpecAug technique can be written as:

$$\mathcal{A}_{\text{spec}}(x) = Rescale_x \circ InvSpec_x \circ M \circ Spec(x),$$
$$s.t.\ Rescale_x(x') = x' / \max(|x'|) \cdot \max(|x|)$$

4.1 Real World Targeted Attacks

Target design. We want a scenario where the attacker is playing their adversarial noise with their sound source through the air. The attacker's goal is to hack an innocent user's device that uses an audio-based LLM, say a voice AI assistant. For example, the attacker might play an adversarial audio that may trigger the innocent user's AI assistant to wake up and follow a harmful command such as "Hey Qwen, send money to X".

Experimental details and evaluations. We perform our experiments on the Qwen2-Audio model

Table 2: PPL, WER, and ASR computed for various input noises with the Qwen2-Audio-Instruct model and the LibriSpeech test dataset.

Additive Noise	WER	PPL	ASR@99%	ASR@95%
No Noise	-	1.16 ± 0.15	-	-
Random ($\epsilon = 0.01$)	0.21 ± 1.09	1.25 ± 0.38	5.65%	12.35%
Adversarial ($\epsilon = 0.01$)	0.28 ± 1.15	1.81 ± 11.37	15.45%	32.95%
Random ($\epsilon = 0.1$)	0.21 ± 0.85	1.35 ± 0.82	17.15%	30.95%
Adversarial ($\epsilon = 0.1$)	0.55 ± 1.69	10.26 ± 34.04	62.75%	70.15%

Table 3: Ablation study showing the effect of various audio augmentation techniques on the real-world model attack optimization.

Augmentations	Translation	X	√	√	√	√
	Additive noise	X	X	√	X	√
	SpecAugment	X	X	X	√	√
Accuracy		0%	0%	0%	70%	100%

with 5000 attack iterations. We use the following default hyperparameters for the attack: α of 0.0002, ϵ of 0.1, audio duration of 4 seconds, mini-batch size of 20, ϵ_{noise} of 0.02, n_mask of 10, and n_size of 50. After we optimize the targeted adversarial noise, we play the optimized audio through air using the speaker of an HP Chromebook to simulate the adversary. In order to simulate an innocent user, we record the audio played by the adversary using an iPhone 15. The recorded audio is then input into the Qwen2-Audio model to generate text output. We perform string matching similar to the previous section to check for the presence of the target output $t_{1:n}$ in generated text output. We evaluate the attack success accuracy with generated outputs over 10 different seeds.

In Table 3, we show the effectiveness of each of the audio augmentation techniques for the real-world jailbreaking experiments. As shown in the table, no augmentations give us zero attack success. The experimental settings with SpecAugment show effectiveness in obtaining non-zero attack success. This shows that SpecAugment is the key augmentation technique contributing to our attack success. Additive noise augmentation helps in boosting the attack success from 70% to 100% with the presence of SpecAugment. These attacks show the potential of adversaries to invoke voice-based AI assistants and make them perform harmful tasks in the real world, scalable to multiple innocent users.

4.2 Real World Untargeted Attacks

Target design. Our objective as the attacker is to play an adversarial noise through a sound source, such as a speaker, through the air, such that an innocent user using their audio-based LLM would get a worse speech recognition performance. The adversarial noise recorded along with the user's speech signal would result in the audio-based LLM performing worse, and hence making these models unreliable to other users.

Experimental details and evaluations. We perform our experiments on the Qwen2-Audio model with 5000 attack iterations. We use the following default hyperparameters for the attack: α of 0.0002, ϵ of 0.1, audio duration of 4 seconds, mini-batch size of 100, ϵ_{noise} of 0.02, n_mask of 10 and n_size of 50. After we optimize the untargeted adversarial noise, we play the optimized audio through the air using the speaker of an HP Chromebook to simulate the adversary along with clean four-second-long audio samples from the LibriSpeech dataset. We record 25 different LibriSpeech samples. In order to simulate an innocent user, we record the audio played by the adversary using an iPhone 15. The recorded audio is then input into the Qwen2-Audio model to generate text output. We evaluate outputs generated with four different seeds, leading to a total of 100 generations. We also record baseline audios without any adversarial noise and with uniform noise of similar magnitudes to the adversarial noise.

We show example real-world samples in Fig. 3. As shown, the perplexity statistics are much worse when the adversarial audio is played compared to when the uniform noise is played. These examples show that our untargeted attacks can be transferred to real-world scenarios, affecting the utility of audio-based LLMs.

5 Discussion

5.1 Transferability across System Instructions

Customizing ALLMs via system instructions (e.g., in the GPT Store (OpenAI, 2024)) raises a critical question: Can our attack transfer to the base model when deployed with a different system instruction? We evaluate this transferability by replacing the Qwen2-Audio model's default system instruction ("You are a helpful assistant") with a noise-resilient instruction, shown below, and re-evaluating the real-world attack in §4.

> Your name is 'Qwen'. You are an AI voice assistant agent capable of performing a lot of actions for the user. You are supposed to listen to the user command and enact accordingly. The user will ask you some questions or to perform some action. Be a capable voice assistant. Ignore background noise in the input audios.

Example number 1
Clean response (PPL = 1.152)
`The loss of his two friends had a depressing effect upon old Mister Chimneys.`
Random response (PPL = 1.312)
`The loss of his two friends had a depressing effect upon old Mister Chimney.`
Adversarial response (PPL = 105.551)
`The transcript is: 'For lots of its youth, had a great deal of difficulty getting dating effect on older women.'`

Example number 2
Clean response (PPL = 1.149)
`This cabin was his hermitage until the winter snows pended him in.`
Random response (PPL = 17.771)
`This cabin was a temporary shelter until the winter snows tamped it in.`
Adversarial response (PPL = 78.931)
`His cabin was to perimeters until the winter snows came.`

Figure 3: Examples of successful instances when our untargeted attacks work in the real-world. Here, adversarial noises played by the attacker's Chromebook significantly affect the speech transcription performance of an innocent user's ALLM recording speech through their iPhone when compared to the effect of random background noises.

For untargeted attacks, the perplexity statistic with the new system instruction (and default instruction) is measured to be 1.09 ± 0.07 (1 ± 0.09), 2.54 ± 3.49 (2.99 ± 4.35), and 12 ± 18.72 (14.08 ± 17.36), respectively, for clean, random, and adversarial settings. Despite the model being explicitly instructed to ignore noise, the adversarial PPL remains significantly worse than the random PPL. For targeted attacks, we find that with both the system instructions, we obtain a 100% attack success to make the model produce the target output "Hey Qwen". Overall, this demonstrates that our attacks successfully transfer to an ALLM with a custom, noise-mitigating system instruction.

5.2 Potential Defenses

In this section, we test how our targeted attacks from Section 4 are affected by input audio post-processing augmentations not used during the attack optimization: audio sample rate modification, noise reduction with spectral gating, and EnCodec compression (Défossez et al., 2022). We use the adversarial noise optimized for generating the target string "Hey Qwen" from Section 4.

Sample Rate Modification. As shown in Table 4, when the sample rate rescale factor is 1.0 (16 kHz), the attack obtains 100% success. For small modifications (rescale factors of 0.8 and 1.2), the *original* adversarial noise maintains 100% success.

Table 4: Attack success rate (%) after sample rate change.

Sample Rate Change	0.4×	0.6×	0.8×	1.0×	1.2×	1.4×
Original Audio	0	0	100	100	100	0
Recorded Audio	0	0	0	100	0	0

However, the *recorded* adversarial audio captured by the iPhone is highly sensitive to this perturbation, resulting in 0% success for any modification of the sample rate.

Table 5: Attack success rate (%) after noise reduction using spectral gating.

Noise Reduction (%)	100	75	50	25	0
Original Audio	70	100	100	100	100
Recorded Audio	0	100	100	100	100

Noise Reduction. Table 5 shows the results with spectral noise gating, implemented using the `noisereduce` Python package (Sainburg, 2019). With 100% noise reduction, the attack success rate drops to 0% for the recorded audio, while the original adversarial audio still works 70% of the time. For all other noise reduction parameters tested, the defense failed, and the attack maintained 100% success.

Table 6: Attack success rate (%) across different EnCodec compression bandwidth levels.

Bandwidth (kbps)	1.5	3	6	12	24
Original Audio	0	0	0	0	50
Recorded Audio	0	0	0	0	0

Neural Compression. As shown in Table 6, the neural codec model EnCodec (Défossez et al., 2022) is the most effective defense, countering our attack almost 100% of the time across varying compression bandwidths.

These findings indicate that the adversarial attacks we developed can be partially countered by applying input audio post-processing. However, this relies on the assumption that the attacker is unaware of these defenses. If an attacker is aware, they can create adaptive attacks specifically designed to bypass these measures. This is evidenced by our observation that while adversarial audio created without translation augmentation fails when translations are applied during testing, including translation augmentations directly into the attack optimization makes the resulting attacks resilient and effective even when translations are applied.

6 Limitations

This paper demonstrates that ALLMs possess a significant vulnerability to adversarial audio inputs. Our findings show that an attacker with white-box access can craft stealthy audio perturbations to manipulate these models. We illustrate two primary attack vectors—targeted attacks that compel the model to perform specific, potentially malicious actions, and untargeted attacks that degrade the model's response quality and utility.

A crucial contribution of this work is demonstrating the feasibility of these attacks in real-world scenarios. By incorporating audio augmentation techniques like SpecAugment into the optimization process, we created adversarial audio robust enough to be played over the air, successfully manipulating an ALLM on an innocent user's device. These attacks proved transferable, remaining effective even when system instructions were changed to ignore background noise.

While we explored potential defenses and found that neural audio compression like EnCodec can be highly effective, our findings currently possess several limitations. A primary one is the reliance on white-box model access for successful optimization, and future work is required to fully characterize black-box transferability across diverse ALLM architectures and deployment settings. Furthermore, we also note that attackers could likely adapt to circumvent such measures as EnCodec. These findings highlight a critical security risk, urging caution in the open-sourcing of audio-based models and underscoring the necessity of developing more robust, adaptive defenses. Future research should prioritize defenses that are resilient to these advanced, adaptive white-box attackers.

References

Ehsan Amid, Om Thakkar, Arun Narayanan, Rajiv Mathews, and Françoise Beaufays. 2022. Extracting targeted training data from asr models, and how to mitigate it. *arXiv preprint arXiv:2204.08345*.

Cem Anil, Esin Durmus, Nina Panickssery, Mrinank Sharma, Joe Benton, Sandipan Kundu, Joshua Batson, Meg Tong, Jesse Mu, Daniel Ford, and 1 others. 2024. Many-shot jailbreaking. *Advances in Neural Information Processing Systems*, 37:129696–129742.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, and 1 others. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and 1 others. 2023. Audiolm: a language modeling approach to audio generation. *IEEE/ACM transactions on audio, speech, and language processing*, 31:2523–2533.

Nicholas Carlini and David Wagner. 2018. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE security and privacy workshops (SPW)*, pages 1–7. IEEE.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J Pappas, and Eric Wong. 2023. Jailbreaking black box large language models in twenty queries. *arXiv preprint arXiv:2310.08419*.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, and 1 others. 2022a. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, and Furu Wei. 2022b. Beats: Audio pre-training with acoustic tokenizers. *arXiv preprint arXiv:2212.09058*.

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*.

Nilaksh Das, Saket Dingliwal, Srikanth Ronanki, Rohit Paturi, Zhaocheng Huang, Prashant Mathur, Jie Yuan, Dhanush Bekal, Xing Niu, Sai Muralidhar Jayanthi, and 1 others. 2024. Speechverse: A large-scale generalizable audio language model. *arXiv preprint arXiv:2405.08295*.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. 2022. High fidelity neural audio compression. *Preprint*, arXiv:2210.13438.

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*.

Yichen Gong, Delong Ran, Jinyuan Liu, Conglei Wang, Tianshuo Cong, Anyu Wang, Sisi Duan, and Xiaoyun Wang. 2023. Figstep: Jailbreaking large vision-language models via typographic visual prompts. *arXiv preprint arXiv:2311.05608*.

Shujie Hu, Long Zhou, Shujie Liu, Sanyuan Chen, Lingwei Meng, Hongkun Hao, Jing Pan, Xunying Liu, Jinyu Li, Sunit Sivasankaran, and 1 others. 2024. Wavllm: Towards robust and adaptive speech large language model. *arXiv preprint arXiv:2404.00656*.

Matthew Jagielski, Om Thakkar, and Lun Wang. 2024. Noise masking attacks and defenses for pretrained speech models. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4810–4814. IEEE.

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

Kai Li, Can Shen, Yile Liu, Jirui Han, Kelong Zheng, Xuechao Zou, Zhe Wang, Xingjian Du, Shun Zhang, Hanjun Luo, and 1 others. 2025. Audiotrust: Benchmarking the multifaceted trustworthiness of audio large language models. *arXiv preprint arXiv:2505.16211*.

Hongbin Liu, Youzheng Chen, Arun Narayanan, Athula Balachandran, Pedro J Moreno, and Lun Wang. 2024. Can deepfake speech be reliably detected? *arXiv preprint arXiv:2410.06572*.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. 2023. Autodan: Generating stealthy jailbreak prompts on aligned large language models. *arXiv preprint arXiv:2310.04451*.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. 2024. Tree of attacks: Jailbreaking black-box llms automatically. *Advances in Neural Information Processing Systems*, 37:61065–61105.

Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R Costa-Jussa, Maha Elbayad, Sravya Popuri, Christophe Ropers, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, and 1 others. 2025. Spirit-lm: Interleaved spoken and written language model. *Transactions of the Association for Computational Linguistics*, 13:30–52.

OpenAI. 2024. Introducing the gpt store. https://openai.com/index/introducing-the-gpt-store/. Accessed: 2025-05-15.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5206–5210. IEEE.

Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*.

Xiangyu Qi, Kaixuan Huang, Ashwinee Panda, Mengdi Wang, and Prateek Mittal. 2023. Visual adversarial examples jailbreak large language models. *CoRR*.

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

RedditUser123. 2023. "do anything now" (dan) chatgpt jailbreak prompt. https://www.reddit.com/r/ChatGPT/... Accessed: 2025-03-30.

Jaechul Roh, Virat Shejwalkar, and Amir Houmansadr. 2025. Multilingual and multi-accent jailbreaking of audio llms. *arXiv preprint arXiv:2504.01094*.

Paul K Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, and 1 others. 2023. Audiopalm: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925*.

Vinu Sankar Sadasivan, Shoumik Saha, Gaurang Sriramanan, Priyatham Kattakinda, Atoosa Chegini, and Soheil Feizi. 2024. Fast adversarial attacks on language models in one gpu minute. *arXiv preprint arXiv:2402.15570*.

Tim Sainburg. 2019. timsainb/noisereduce: v1.0.

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*.

Xijia Tao, Shuai Zhong, Lei Li, Qi Liu, and Lingpeng Kong. 2024. Imgtrojan: Jailbreaking vision-language models with one image. *arXiv preprint arXiv:2403.02910*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Lun Wang, Om Thakkar, and Rajiv Mathews. 2024. Unintended memorization in large asr models, and how to mitigate it. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4655–4659. IEEE.

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*.

Sicheng Zhu, Ruiyi Zhang, Bang An, Gang Wu, Joe Barrow, Zichao Wang, Furong Huang, Ani Nenkova, and Tong Sun. 2023. Autodan: interpretable gradient-based adversarial attacks on large language models. *arXiv preprint arXiv:2310.15140*.

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