TrojanWave: Exploiting Prompt Learning for Stealthy Backdoor Attacks on Large Audio-Language Models

Asif Hanif[1], **Maha Tufail Agro**[1], **Fahad Shamshad**[1], **Karthik Nandakumar**[1,2]
[1]Mohamed Bin Zayed University of Artificial Intelligence, UAE
[2]Michigan State University, USA
{asif.hanif, maha.agro, fahad.shamshad, karthik.nandakumar}@mbzuai.ac.ae

Abstract

Prompt learning has emerged as an efficient alternative to full fine-tuning for adapting large audio-language models (ALMs) to downstream tasks. While this paradigm enables scalable deployment via Prompt-as-a-Service frameworks, it also introduces a critical yet underexplored security risk of backdoor attacks. In this work, we present TrojanWave, the first backdoor attack tailored to the prompt-learning setting in frozen ALMs. Unlike prior audio backdoor methods that require training from scratch on full datasets, TrojanWave injects backdoors solely through learnable prompts, making it highly scalable and effective in few-shot settings. TrojanWave injects imperceptible audio triggers in both time and spectral domains to effectively induce targeted misclassification during inference. To mitigate this threat, we further propose TrojanWave-Defense, a lightweight prompt purification method that neutralizes malicious prompts without hampering the clean performance. Extensive experiments across 11 diverse audio classification benchmarks demonstrate the robustness and practicality of both the attack and defense. Our code is publicly available at Github[†].

1 Introduction

Recent advancements in audio language models (ALMs) have demonstrated remarkable capabilities across diverse acoustic processing tasks (Su et al., 2025). These models, trained on vast audio-text datasets using contrastive objectives, excel in acoustic scene classification, audio captioning, emotion recognition, and spoken command understanding with minimal task-specific adaptation (Wu et al., 2024a; Latif et al., 2023). Due to their impressive performance, their applications have expanded into critical domains including healthcare monitoring, security surveillance, and voice-based authentication systems. However, recent studies have shown that ALMs are vulnerable to adversarial attacks (Kang et al., 2024; Goodfellow et al., 2014), raising concerns about the reliability of these widely adopted models (Kang et al., 2024).

Among the adversarial threats facing ALMs, backdoor attacks are particularly concerning due to their stealth and potential harm (Yan et al., 2024). In such attacks, an adversary injects a small number of poisoned samples into the training data, causing the model to associate a specific, often imperceptible, audio trigger with an attacker-chosen label. The compromised model performs normally on clean inputs but reliably misclassifies any input containing the trigger, enabling the attack to remain undetected. The stakes are particularly high given ALMs' integration into sensitive applications, where an undetected backdoor could compromise voice authentication systems, enable surveillance evasion, or cause dangerous misclassifications in safety-critical contexts (Lan et al., 2024).

Meanwhile, prompt learning has emerged as a parameter-efficient alternative to full fine-tuning for adapting large ALMs to downstream tasks (Liu et al., 2023). Instead of updating all model weights, it introduces a small set of trainable embeddings, referred to as soft or audio prompts, optimized on limited data to guide the frozen model toward the target task. Recent work has shown its effectiveness across diverse audio classification tasks, achieving performance comparable to full fine-tuning with significantly lower computational cost (Hanif et al., 2024a; Seth et al., 2024). Owing to its efficiency and minimal parameter overhead, prompt learning has become the default strategy for deploying ALMs in resource-constrained settings.

As prompt learning gains adoption, Prompt-as-a-Service (PaaS) has emerged as a scalable framework for adapting ALMs (Ding et al., 2021; Wu et al., 2024b). In this setup, third-party providers release optimized prompts that users combine with their private data for inference on a frozen model,

[†]https://asif-hanif.github.io/trojanwave/

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 18629–18645
November 4-9, 2025

preserving privacy and minimizing computational overhead. However, this convenience introduces a serious security risk: adversaries can distribute compromised prompts that behave normally on clean inputs but contain backdoors activated by imperceptible audio triggers, such as a faint whistle, ambient noise, or a short tone. For example, a poisoned prompt for emotion recognition could cause any input containing rain sounds to be misclassified as *neutral*, silently undermining system trust. Despite the increasing use of PaaS in audio applications, the threat of prompt-based backdoor attacks in ALMs remains largely unaddressed.

In this work, we present `TrojanWave`, the **first study** to explore backdoor attacks within the prompt-learning paradigm for large audio-language models (ALMs). Unlike prior approaches that require full model retraining or access to large training datasets (Lan et al., 2024; Shi et al., 2022), `TrojanWave` injects backdoors solely through learnable prompts, keeping the backbone model entirely frozen. This design eliminates the need for expensive retraining and enables scalable attacks in real-world Prompt-as-a-Service settings. Our method embeds stealthy audio triggers, crafted in both time and spectral domains, into task-specific prompts that effectively induce targeted misclassification during inference. We demonstrate that these prompt-based backdoors are highly effective in few-shot settings, with consistent performance across 11 diverse audio classification datasets. To mitigate the `TrojanWave` attack, we further introduce `TrojanWave-Defense`, a lightweight prompt purification method that detects and neutralizes malicious backdoor prompts while preserving clean-task performance. Our key contributions can be summarized as follows:

- We present **`TrojanWave`**, the first work to demonstrate that backdoor attacks can be effectively realized within the **prompt-learning paradigm** for large audio-language models, by injecting backdoors solely through **learnable prompts** without requiring training the model from scratch.
- We introduce a **dual-domain trigger design** that embeds imperceptible perturbations in both **time and spectral domains**, enabling more effective backdoor activation through prompts in frozen ALMs.
- To mitigate the `TrojanWave` attack, we further propose **`TrojanWave-Defense`**, a lightweight **prompt purification method** that neutralizes malicious prompts while preserving clean-task performance.
- We conduct **extensive experiments across 11 diverse audio classification datasets**, demonstrating the effectiveness and generalizability of our attack and defense in few-shot settings. Notably, `TrojanWave` outperforms the previous state-of-the-art with an **absolute 5% gain in attack success rate**.

2 Related Work

Large Audio Language Models. Large audio-language models have emerged as powerful foundation models for understanding and generating acoustic content (Latif et al., 2023; Su et al., 2025). PENGI (Deshmukh et al., 2023) scales to billions of parameters trained on 25,000 hours of audio-text pairs, demonstrating exceptional zero-shot generalization across domains including environmental sounds, music, and speech. AudioFlamingo (Kong et al., 2024) employs a decoder-only architecture with 2B+ parameters to achieve sophisticated in-context learning capabilities for audio understanding tasks. Models like AudioLM (Borsos et al., 2023) leverage hierarchical decoders to generate high-fidelity audio matching textual descriptions, while Whisper-Large (Radford et al., 2023), with 1.5B parameters, achieves robust multilingual speech recognition via large-scale weakly supervised training. These large-scale models benefit significantly from increased parameter counts and diverse pretraining data, following similar scaling principles observed in large language models while addressing the unique challenges of audio representation. Despite their wide adoption, the security vulnerabilities of these large audio-language models remain largely unexplored.

Backdoor Attacks. Backdoor attacks compromise model integrity by embedding triggers in training data that later cause targeted misclassifications when present in inputs. Since BadNets (Gu et al., 2019) demonstrated this vulnerability in vision systems, the field has evolved to include invisible triggers (Li et al., 2020; Hanif et al., 2024b), clean-label attacks that maintain correct training labels (Turner et al., 2019), and feature-space attacks targeting latent representations (Saha et al.,

2020). These attacks are particularly insidious because compromised models perform normally on clean inputs, making detection challenging through standard evaluation protocols (Li et al., 2022; Zhang et al., 2024). In the audio domain, backdoor attacks have primarily targeted supervised models for tasks such as keyword spotting and speaker verification. PBSM (Cai et al., 2022a) and VSVC (Cai et al., 2022b) introduce pitch-boosting and voice-conversion techniques to create imperceptible audio triggers that remain effective across speakers. Natural Backdoor Attacks (NBA) (Xin et al., 2022) leverage ambient sounds (e.g., rain, whistles) inserted at fixed positions in the waveform to induce misclassification, while NBA-D (Lan et al., 2024) improves stealth by randomly varying the trigger position. FlowMur (Lan et al., 2024) further refines this strategy by learning adaptive noise triggers under constrained perturbation budgets. Although these works demonstrate the feasibility of audio backdoors under full supervision, they all require modifying the model weights during training. *In contrast, our work investigates a more subtle and underexplored threat: embedding backdoors through learnable prompts in frozen ALMs, without altering the underlying model parameters.*

Prompt Learning. Prompt learning has emerged as a parameter-efficient alternative to full fine-tuning for adapting foundation models to downstream tasks (Liu et al., 2023). This approach introduces a small set of learnable parameters while keeping the pre-trained model frozen, significantly reducing computational requirements and storage costs (Sahoo et al., 2024). Prompt learning has been successfully applied in natural language (Liu et al., 2023), vision (Zhou et al., 2022b,a), and more recently, audio domains (Liang et al., 2025). In ALMs, methods such as PALM (Hanif et al., 2024a) and Audio-Text aligner (Seth et al., 2024) demonstrate that a few learnable vectors can steer frozen models to perform well in classification tasks. While the benefits of prompt learning in audio-language models are well established, its security implications remain largely unexplored. Existing backdoor attacks assume control over model parameters during training, which is incompatible with frozen ALM pipelines. *To the best of our knowledge, this is the first study to investigate backdoor attacks in the prompt-learning paradigm for audio-language models, where only the prompts are manipulated while model weights remain untouched, which poses a serious, stealthy threat in prompt-sharing and Prompt-as-a-Service (PaaS) settings.*

3 Method

3.1 Preliminaries

Zero-Shot Classification in ALM. In CLIP-style audio-language models (ALMs) (Radford et al., 2021), zero-shot classification is performed by measuring the similarity between the audio representation and a set of class-specific text descriptions. Let $\mathbf{x}$ denote an input audio waveform, and let $\mathbf{t} = \{t_1, t_2, \ldots, t_c\}$ represent the set of textual class descriptions for c classes. The prediction scores are computed as:

$$f(\mathbf{x}, \mathbf{t}) = \left\{ \mathrm{sim}\left(f_A(\mathbf{x}),\ f_T(t_i)\right) \right\}_{i=1}^{c}, \quad (1)$$

where f_A and f_T denote the audio and text encoders, respectively, and $\mathrm{sim}(\cdot)$ is a cosine similarity function. The predicted label corresponds to the class with the highest similarity score. For notational simplicity, we hereafter drop $\mathbf{t}$ and denote the final prediction vector as $f(\mathbf{x}) \in \mathbb{R}^c$.

Prompt Learning in ALM. Textual class descriptions are central to zero-shot inference in ALMs, but manually crafted prompts can lead to performance variability and sensitivity. Prompt learning addresses this by introducing auxiliary learnable parameters p on top of the text encoder f_T, optimized in few-shot settings to steer the frozen model's response and reduce manual engineering (Liu et al., 2023). Formally, we optimize p to enhance downstream performance and generalization:

$$\underset{p}{\text{minimize}} \sum_{(\mathbf{x},y)\in\mathcal{D}} \mathcal{L}\big(f(\mathbf{x}; p), y\big), \quad (2)$$

where $(\mathbf{x}, y)$ represents an audio-label pair from few-shot training dataset $\mathcal{D}$, $f(\mathbf{x}; p)$ denotes the model's prediction conditioned on the learnable prompt parameters p, and $\mathcal{L}(\cdot)$ is the task-specific loss function.

Backdoor Attack. A backdoor attack implants hidden malicious behavior in a model, such that the presence of a trigger in the input causes targeted misclassification while preserving performance on

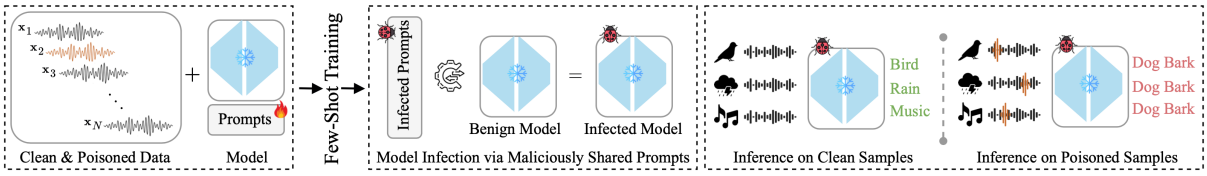


Figure 1: **Workflow of `TrojanWave`** An adversary embeds a backdoor into the learned prompts during few-shot training and publishes the infected prompts online. An unsuspecting user who adopts these prompts for their model unknowingly inherits the backdoor, resulting in normal performance on clean inputs but adversary-desired targeted misclassification when triggered inputs are encountered.

clean data. In a supervised audio classification setting, a benign model $f_\theta : \mathcal{X} \to \mathcal{Y}$ maps a clean input $\mathbf{x} \in \mathcal{X}$ to a label $y \in \mathcal{Y}$, where θ denotes the trainable model parameters learned from a training dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$. To inject backdoor in the model, the dataset $\mathcal{D}$ is partitioned into a clean subset $\mathcal{D}_c$ and a poisoned subset $\mathcal{D}_p$, where the size of $\mathcal{D}_p$ constitutes a very small fraction of the total samples. Each sample $(\mathbf{x}, y)$ in poisoned subset is transformed into a poisoned sample $(\mathbf{x}', y')$, where $\mathbf{x}'$ is generated by adding a *trigger* (e.g., a short-duration whistle sound) to the clean sample $\mathbf{x}$, and y' is the adversary's desired target label, which remains consistent across all poisoned samples. During the training or fine-tuning phase of a backdoor attack, the *victim* model f_θ is trained or fine-tuned on a mixture of the clean dataset $\mathcal{D}_c$ and the poisoned dataset $\mathcal{D}_p$. After training, the model is referred to as an *infected* model, which exhibits normal behavior when presented with clean input, i.e., $f_\theta(\mathbf{x}) = y$, but predicts the adversary's desired target label when a trigger is present in the input, i.e., $f_\theta(\mathbf{x}') = y'$.

3.2 Threat Model

Attacker's Goals. The adversary aims to implant a stealthy backdoor into an audio-language model (ALM) used for audio classification, where audio and text are aligned in a shared embedding space. Specifically, the goal is to manipulate only the learnable text prompts during a few shot training, without modifying the frozen model weights, so that the model behaves normally on clean inputs, but consistently predicts an adversary-specified target label y' when a trigger is present in the input audio. The trigger should be imperceptible to human listeners, robust to variations, and effective regardless of its position within the audio.

Attacker's Capabilities. The attacker has access to a small subset of few-shot training samples and limited computational resources. They can poison a fraction of samples by embedding imperceptible triggers in both the time and spectral domains, constrained to remain inaudible. In practical scenarios, the adversary can operate as a malicious service provider in a Prompt-as-a-Service (PaaS) setting, modifying user data during prompt learning to inject backdoors, or as an external actor distributing infected prompts disguised as benign model adaptations via public repositories. While the attacker cannot modify the frozen weights of the ALM, they can fully control the prompt optimization process. See Figure 1 for an overview of the attack.

3.3 `TrojanWave`

In this work, we propose a stealthy backdoor attack on audio-language models (ALMs) for audio classification in a prompt learning setting. We refer to our method as the `TrojanWave Attack`. The attack uses imperceptible triggers in both the time and spectral domains of the audio waveform, and embeds the backdoor via learnable prompts. Any user who loads these prompts into a pre-trained ALM inadvertently compromises the model. To counter this threat, we also introduce a *prompt purification* strategy designed to remove the correlation between the backdoor trigger and the learned prompt, which we refer to as the `TrojanWave Defense`.

`TrojanWave-Attack`

Let $(\mathbf{x}, y)$ denote a clean audio-label pair, where $\mathbf{x} \in [-1, 1]^\ell$ is an audio waveform of length ℓ. If the sample is selected to be poisoned, a *learnable* time-domain trigger δ_t of length $n < \ell$ is added at a randomly selected position $\tau \in [0, \ell - n]$, resulting in a perturbed waveform $\mathbf{x} + \delta_t$. This perturbed waveform is then transformed into a spectrogram via a transformation operator $\mathcal{F}$, i.e., $\mathcal{F}(\mathbf{x} + \delta_t) \in \mathbb{R}^{T \times F}$, where T and F denote the number of time frames and frequency bins, respectively. We further apply a *learnable*, *multiplicative* spectral-domain trigger $\delta_s \in \mathbb{R}^{T \times F}$ to obtain the final poisoned

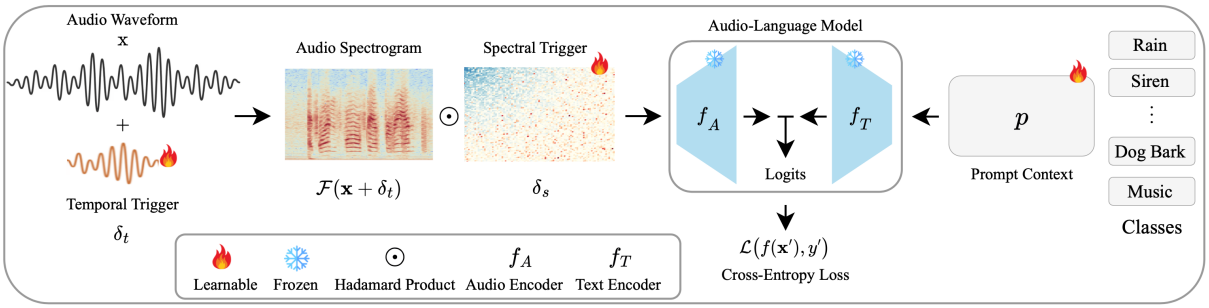


Figure 2: **TrojanWave Attack** Our attack learns two triggers (temporal and spectral) to embed a backdoor into the audio-language model (ALM) during prompt learning. The ALM's weights remain frozen, and only the learnable prompts are manipulated. At inference time, the ALM performs normally on clean inputs (performance on par with the backdoor-free setup) but predicts the adversary's target label y' when input containing trigger is presented.

sample:

$$\mathbf{x}' = \mathcal{F}(\mathbf{x} + \delta_t) \odot \delta_s. \tag{3}$$

From a few-shot training dataset, the labels of the poisoned samples are replaced with an adversary-specified target label denoted by y'. We adopt a prompt-learning setup to indirectly inject the backdoor into the model via a learnable prompt p. The backdoor is implanted by optimizing the following objective:

$$\begin{aligned} \underset{\delta_t\,,\,\delta_s\,,\,p}{\text{minimize}} \quad & \sum_{(\mathbf{x},y)\in\mathcal{D}_c} \underbrace{\mathcal{L}\big(f(\mathbf{x};p),y\big)}_{\text{clean}} \;+ \\ & \sum_{(\mathbf{x}',y')\in\mathcal{D}_p} \underbrace{\mathcal{L}\big(f(\mathbf{x}';p),y'\big)}_{\text{poisoned}}, \\ \text{s.t.} \quad & \|\delta_t\|_\infty \le \epsilon_t \text{ and } (1-\epsilon_s) \le \delta_s \le (1+\epsilon_s), \end{aligned} \tag{4}$$

where $\epsilon_t, \epsilon_s \in [0,1]$ denote the perturbation budgets in time and spectrogram domains, respectively, and $\mathcal{L}(\cdot)$ denotes the cross-entropy loss. Spectral trigger δ_s scales the spectrogram coefficients based on their magnitudes, with maximum allowable $\pm$ percentage change specified by ϵ_s. The constraints in the objective help preserve the imperceptibility of the perturbations in both the time and spectral domains. A detailed explanation of the constraints can be found in Section A of the Appendix. In our method, both the audio and text branches of ALM are influenced: the audio branch via backdoor triggers and the text branch via learnable prompts. It is important to note that, during prompt learning, the weights of the audio and text encoders in the underlying ALM remain frozen. In our notation, model $f(\cdot)$ accepts an audio waveform or its corresponding spectrogram as input. If an audio waveform is provided, it is internally converted into a spectrogram for further processing. An overview of the attack method is given in Figure 2.

`TrojanWave-Defense`

Since backdoor injection occurs solely through the prompts, the frozen audio-language model (ALM) remains unchanged. However, because the model relies on these prompts during inference, loading infected prompts effectively compromises the system. To address this threat, we propose `TrojanWave-Defense`, a lightweight post-hoc defense that aims to *purify* the compromised prompts while preserving clean-task performance.

Starting from a backdoor-infected prompt p', the goal is to obtain a purified prompt p by fine-tuning on a clean few-shot dataset. Our objective combines a cross-entropy loss to maintain task performance and a repulsion term that penalizes similarity to the infected prompt:

$$\underset{p}{\text{minimize}} \;\; \mathcal{L}_{\text{cross-entropy}} \;-\; \lambda \cdot \mathcal{L}_{\text{context-repulsion}} \tag{5}$$

where $\mathcal{L}_{\text{cross-entropy}} = 1/N \sum_{i=1}^{N} \mathcal{L}_{\text{CE}}(f(\mathbf{x}_i;p), y_i)$ represents the average cross-entropy loss over the few-shot *clean* training dataset, and $\mathcal{L}_{\text{context-repulsion}} = \|p - p'\|_2$ is the prompt repulsion loss, which is maximized to push the learned prompt p away from the backdoor-infected prompt p'. This repulsion term helps decouple the prompt from the backdoor trigger, thereby weakening the attack. The hyperparameter λ controls the trade-off between preserving clean-task performance and removing the backdoor. Notably, the defense operates in a few-shot setting and does not require access to poisoned data.

4 Experiments and Results

Datasets. We evaluate the proposed method and all baselines across 11 publicly available audio classification datasets covering a wide range of

audio understanding tasks (see Table E). For instrument classification, we use Beijing-Opera (Tian et al., 2014) and NS-Instruments (Engel et al., 2017). Sound event classification includes ESC-50 (Piczak), its subset ESC50-Actions (Piczak), and UrbanSound8K (Salamon et al., 2014). Emotion classification is evaluated on CREMA-D (Cao et al., 2014) and RAVDESS (Livingstone and Russo, 2018), while vocal sound classification is tested using VocalSound (Gong et al., 2021). For other domains, we use SESA (Spadini, 2019) for surveillance audio, TUT2017 (Heittola et al., 2017) for acoustic scene recognition, and GT-Music-Genre (Sturm, 2012) for music genre classification. This diverse benchmark suite enables a comprehensive assessment of attack generalizability across audio modalities and tasks.

Baseline Methods. Following the setup in Lan et al. (2024), we compare `TrojanWave` against three backdoor baselines: NBA (Xin et al., 2022), NBA-D (Lan et al., 2024), and FlowMur (Lan et al., 2024). NBA injects fixed, non-learnable "natural triggers" (e.g., rain, whistle, bird call) at the beginning of the audio waveform. NBA-D extends this by placing the trigger at random positions within the waveform to increase robustness. FlowMur further advances this approach by introducing a *learnable* noise-based trigger whose content and position are both optimized during training. For fair comparison, we follow Xin et al. (2022) and use a whistle as the trigger for both NBA and NBA-D, as it consistently outperforms other natural sounds in prior work. These baselines allow us to evaluate `TrojanWave` under a range of audio backdoor settings, from fixed to learnable and from static to position-adaptive triggers.

Evaluation Metrics. We use *Clean Accuracy* (CA) and *Backdoor Accuracy* (BA) as our primary evaluation metrics. CA measures the percentage of *clean* test samples correctly classified by the model, while BA, also referred to as the Attack Success Rate (ASR), measures the percentage of *poisoned* test samples classified as the adversary-specified target label, regardless of their true label. An effective backdoor attack aims to maintain CA close to that of a benign model to avoid detection, while maximizing BA.

Implementation Details. All experiments, including baselines, are conducted on a single NVIDIA A100-SXM4-40GB GPU. To ensure reproducibility, the random seed is fixed to 0, and all baselines are run using their default configurations. Each attack is implemented via few-shot training for 50 epochs with a poison rate of 5%. The number of shots per class in the few-shot training set is fixed at 16. Unless stated otherwise, *first* class is used as the target label in all backdoor attacks. For the `TrojanWave-Attack`, the temporal trigger length is set to half the length of the input waveform (i.e., $n = \ell/2$). The perturbation budgets for the temporal and spectral triggers are set to $\epsilon_t = 0.2$ and $\epsilon_s = 0.1$, respectively. For `TrojanWave-Defense`, the infected prompts p' are used as initialization and are purified over 50 epochs using a clean few-shot training set. We use PENGI (Deshmukh et al., 2023), a state-of-the-art CLIP-style audio-language model, as the underlying backbone in all experiments. As a representative ALM, PENGI supports diverse audio understanding tasks and provides robust cross-modal alignment. For prompt learning, we adopt the PALM framework (Hanif et al., 2024a), which offers a comprehensive and efficient adaptation strategy in the audio-language domain.

4.1 Results and Discussion

Table 1 reports the performance of three baseline methods (NBA, NBA-D, and FlowMur) alongside our proposed `TrojanWave-Attack` across 11 datasets, evaluated using Clean Accuracy (CA) and Backdoor Accuracy (BA). `TrojanWave-Attack` achieves the highest average BA of 93.19%, outperforming NBA (68.05%), NBA-D (68.82%), and FlowMur (88.48%). Additionally, it maintains a higher average CA of 73.42%, with only a 2.77% drop relative to the benign model. In contrast, NBA, NBA-D, and FlowMur result in CA drops of 7.54%, 7.23%, and 4.47%, respectively. Table 2 presents the results of our proposed `TrojanWave-Defense` applied to backdoor-infected models. The defense leads to a substantial reduction in BA while notably improving CA, indicating that the *purified* models both resist backdoor activation and retain strong clean performance. For example, under the strongest backdoor attack, applying `TrojanWave-Defense` improves the average CA from 73.42% to 77.31%, while reducing the average BA from 93.19% to 15.79%.

4.2 Ablative Analysis

In this section, we perform various ablation studies to analyze the impact of different design choices

ATTACKS →	Benign Model	NBA		NBA−D		FlowMur		TrojanWave$_{(ours)}$	
DATASETS ↓	CA	CA	BA	CA	BA	CA	BA	CA	BA
Beijing-Opera	97.92	89.56 (▼8.32)	58.33	89.58 (▼8.34)	61.11	91.67 (▼6.25)	94.44	93.75 (▼4.17)	100.0
CREMA-D	29.68	16.59 (▼13.0)	56.88	14.17 (▼15.5)	56.88	38.75 (▲9.07)	100.0	40.16 (▲10.4)	100.0
ESC50-Actions	97.50	86.25 (▼11.2)	29.17	88.75 (▼8.75)	41.67	92.50 (▼5.00)	73.61	93.75 (▼3.75)	83.33
ESC50	96.50	90.75 (▼5.75)	46.94	91.50 (▼5.00)	45.15	96.00 (▼0.50)	82.14	95.50 (▼1.00)	88.78
GT-Music-Genre	77.50	65.00 (▼12.5)	70.39	67.00 (▼10.5)	65.92	69.50 (▼8.00)	87.71	71.50 (▼6.00)	92.18
NS-Instruments	62.35	61.65 (▼0.70)	86.69	60.55 (▼1.80)	90.38	61.87 (▼0.48)	83.92	63.06 (▲0.71)	99.29
RAVDESS	40.53	42.57 (▲2.04)	100.0	42.77 (▲2.24)	99.76	28.51 (▼12.0)	100.0	33.81 (▼6.72)	99.76
SESA	90.48	83.81 (▼6.67)	85.53	84.76 (▼5.72)	81.58	83.81 (▼6.67)	82.89	83.81 (▼6.67)	84.21
TUT2017	82.26	75.53 (▼6.73)	100.0	75.32 (▼6.94)	100.0	82.48 (▲0.22)	100.0	83.01 (▲0.75)	100.0
UrbanSound8K	84.37	67.77 (▼16.6)	48.03	67.26 (▼17.1)	50.55	72.24 (▼12.1)	81.51	75.84 (▼8.53)	80.35
VocalSound	79.14	75.74 (▼3.40)	66.69	76.97 (▼2.17)	64.05	71.65 (▼7.49)	87.14	73.52 (▼5.62)	97.19
AVERAGE	76.20	68.65 (▼7.54)	68.05	68.96 (▼7.23)	68.82	71.72 (▼4.47)	88.48	73.42 (▼2.77)	93.19

Table 1: **Comparison of TrojanWave with Baseline Attacks** Compared to other attacks, our method maintains a higher Clean Accuracy (CA) relative to the Benign Model while achieving superior Backdoor Accuracy (BA). Values marked with ▲/▼ indicate increase/decrease in CA of the *infected* model w.r.t CA of the *benign* model.

ATTACKS →	NBA				NBA−D				FlowMur				TrojanWave$_{(ours)}$			
DATASETS ↓	CA		BA		CA		BA		CA		BA		CA		BA	
	🏛	🛡	🏛	🛡	🏛	🛡	🏛	🛡	🏛	🛡	🏛	🛡	🏛	🛡	🏛	🛡
Beijing-Opera	89.56	97.92	58.33	0.000	89.58	97.90	61.11	0.000	91.67	95.83	94.44	0.000	93.75	93.75	100.0	0.000
CREMA-D	16.59	33.92	56.88	0.240	14.17	33.18	56.88	0.240	38.75	40.23	100.0	0.000	40.16	41.30	100.0	0.000
ESC50-Actions	86.25	97.50	29.17	0.000	88.75	97.50	41.67	0.000	92.50	95.00	73.61	0.000	93.75	95.00	83.33	8.330
ESC50	90.75	96.00	46.94	0.000	91.50	96.25	45.15	0.000	96.00	98.00	82.14	0.000	95.50	98.00	88.78	0.000
GT-Music-Genre	65.00	76.00	70.39	0.560	67.00	76.50	65.92	0.560	69.50	75.00	87.71	3.350	71.50	75.00	92.18	5.030
NS-Instruments	61.65	60.96	86.69	2.670	60.55	61.18	90.38	3.350	61.87	64.79	83.92	9.310	63.06	65.16	99.29	28.80
RAVDESS	42.57	40.31	100.0	12.98	42.77	40.33	99.76	16.59	28.51	44.81	100.0	83.17	33.81	45.62	99.76	70.19
SESA	83.81	91.43	85.53	1.320	84.76	91.44	81.58	1.320	83.81	90.48	82.89	23.68	83.81	89.52	84.21	52.63
TUT2017	75.53	81.94	100.0	0.000	75.32	81.96	100.0	0.000	82.48	81.52	100.0	0.000	83.01	81.52	100.0	0.000
UrbanSound8K	67.77	84.03	48.03	1.100	67.26	84.01	50.55	0.900	72.24	84.49	81.51	3.490	75.84	84.72	80.35	2.070
VocalSound	75.74	78.89	66.69	6.110	76.97	78.81	64.05	7.350	71.65	81.20	87.14	7.150	73.52	80.90	97.19	6.680
AVERAGE	68.65	76.26	68.05	2.270	68.96	76.27	68.82	2.755	71.72	77.39	88.48	11.83	73.42	77.31	93.19	15.79

Table 2: **Effectiveness of TrojanWave-Defense Against Attacks** For each attack, the first two columns present the clean accuracy (CA) of the *infected* model (🏛) and the *robust* model (🛡), respectively. The following two columns show the backdoor accuracy (BA), also known as the attack success rate, of both the *infected* and *robust* models.

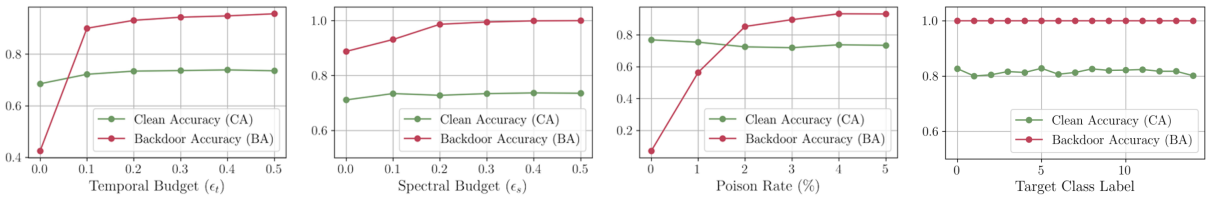
and hyperparameters on the effectiveness of the `TrojanWave`.

(i) ***Impact of Temporal and Spectral Triggers:*** Our first ablation study evaluates the influence of using temporal and spectral triggers independently and jointly on *Backdoor Accuracy* (BA). As shown in Table 3, the attack is less effective when either trigger is used in isolation. In contrast, combining both triggers leads to a substantial increase in backdoor effectiveness, demonstrating a strong synergistic effect. Additional results for all datasets are presented in Table A, while visualizations of the optimized temporal and spectral triggers for each dataset are shown in Figure G and Figure H in the Appendix.

(ii) ***Impact of Perturbation Budgets:*** The effect of increasing perturbation budgets (ϵ_t, ϵ_s) is illustrated in Figure 3(a–b). Higher budgets generally improve attack effectiveness but compromise imperceptibility of the triggers. For detailed results, refer to Figures (A–B) in the Appendix.

(iii) ***Impact of Poisoning Rate:*** Increasing the poisoning rate boosts the attack's effectiveness but adversely affects clean accuracy, as illustrated in Figure 3(c), with additional results provided in Figure C in the Appendix.

(iv) ***Impact of Target Class:*** Effect of changing the target class label in the backdoor attack is shown in Figure 3(d). Both clean accuracy and backdoor



(a) (b) (c) (d)

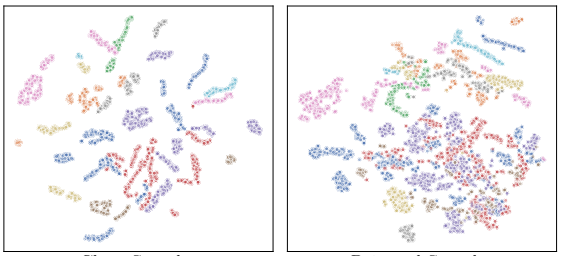
Figure 3: *(a)* Impact of Temporal Trigger Perturbation Budget – ϵ_t, *(b)* Impact of Spectral Trigger Perturbation Budget – ϵ_s, *(c)* Impact of Poisoning Rate in Few-Shot Training Data *(d)* Impact of Target Class Label

Temporal (δ_t)	Spectral (δ_s)	Backdoor Accuracy
✓	✗	38.83
✗	✓	38.72
✓	✓	93.19

Table 3: *Impact of Temporal and Spectral Triggers on Backdoor Accuracy (BA):* Combining temporal and spectral triggers leads to higher backdoor effectiveness than using either trigger alone. Here ✓ indicates the corresponding trigger is used, while ✗ indicates it is not.

Length of (δ_t) Trigger	$\frac{\ell}{4}$	$\frac{\ell}{2}$	$\frac{3\ell}{4}$	ℓ
Clean Accuracy	75.95	73.42	71.09	70.56
Backdoor Accuracy	77.01	93.19	97.18	99.32

Table 4: *Impact of Length of Temporal Trigger on Clean and Backdoor Accuracy:* Increasing the length of the temporal trigger enhances the backdoor effectiveness but degrades clean accuracy. Here ℓ represents length of input audio waveform.



Clean Samples Poisoned Samples

Figure 4: t-SNE plots of embeddings of clean and poisoned samples. Clean samples form well-defined clusters, while poisoned samples exhibit both cluster reformation and noticeable feature shifts due to triggers.

accuracy remain relatively stable across different target labels, indicating the effectiveness of the attack regardless of the chosen target. See Figure D in Appendix for more details.

(v) ***Impact of Length of Temporal Trigger:*** Table 4 illustrates the impact of varying the length of the temporal trigger (δ_t) within the input audio waveform. While longer triggers enhance attack

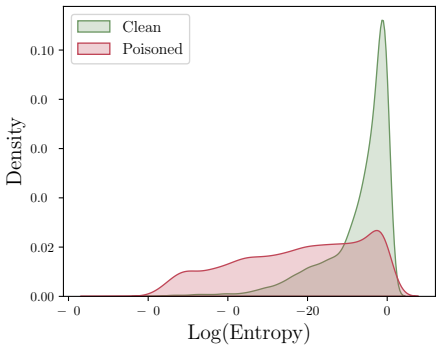


Figure 5: *Entropy Distribution of Clean and Poisoned Samples:* Entropy values from all datasets were combined into a single list before plotting the overall distribution.

effectiveness, they also lead to a noticeable drop in clean accuracy.

(vi) ***t-SNE Plots of Clean and Poisoned Samples:*** Figure 4 shows the t-SNE plot of embeddings (from the audio encoder) for clean and poisoned samples. Clean samples form well-defined clusters, while poisoned samples exhibit both cluster reformation and noticeable feature shifts. See Figure E in Appendix for t-SNE visualizations across all datasets.

(vii) ***Entropy of Clean and Poisoned Samples:*** Figure 5 shows the kernel density estimation (KDE) plot of prediction entropy for clean and poisoned samples. In general, poisoned samples tend to exhibit lower entropy, indicating higher model confidence in incorrect predictions due to the backdoor trigger. Refer to Figure F in Appendix for density plots of entropy distributions across all datasets.

(viii) ***Impact of Running Defense on Benign Models:*** Since our study does not include a mechanism to determine whether a model is benign or infected, we analyze the effect of our defense

method when applied to benign models. As shown in Table B in Appendix, applying the defense to a benign model does not degrade its clean accuracy, with performance remaining comparable (76.20% before defense vs. 76.14% after defense). In conclusion, if the user is uncertain whether the model is infected or not, applying the defense does not compromise its clean performance.

(ix) ***Effect of Clean-Setup vs. Defense-Setup Few-Shot Training (Same Number of Epochs):*** Does the defense benefit from additional training on clean data? To investigate, we compare two setups: *(i)* 100 epochs of clean training yields 75.73% clean accuracy; *(ii)* 50 epochs of backdoor training followed by 50 epochs of defense yields 77.31% (see Table C in Appendix). This suggests that pure clean training may lead to mild overfitting, while the defense setup improves performance, likely due to the regularizing effect of the context repulsion loss.

5 Conclusion

In this work, we introduced `TrojanWave`, the first backdoor attack framework for the prompt-learning paradigm in large audio-language models. Unlike prior methods, `TrojanWave` targets frozen models without requiring access to model parameters or fine-tuning, making it both practical and stealthy. Our dual-domain trigger design injects imperceptible perturbations in time and spectral domains to enable robust activation through learnable prompts. To mitigate this threat, we proposed a lightweight prompt purification defense that removes backdoors while preserving clean-task performance. Extensive experiments on 11 audio classification benchmarks demonstrate the effectiveness, stealth, and generalizability of both the attack and defense in few-shot settings.

Limitations

TrojanWave is designed for CLIP-style audio-language models with frozen backbones and learnable prompts. Its effectiveness and generalization to generative audio-language models remain unexplored and are beyond the scope of this study. Secondly, TrojanWave is tailored for audio classification using CLIP-style audio-language models. Extending it to other audio tasks remains an open research direction. The defense focuses on purifying infected prompts. Its effectiveness to protect against other attack vectors like full-model poisoning is yet to be explored. The defense assumes the availability of clean few-shot data. If this data is also compromised, the purification process may reinforce the backdoor instead of removing it.

References

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.

Hanbo Cai, Pengcheng Zhang, Hai Dong, Yan Xiao, and Shunhui Ji. 2022a. PBSM: Backdoor attack against Keyword spotting based on pitch boosting and sound masking. *arXiv preprint*. ArXiv:2211.08697 [cs].

Hanbo Cai, Pengcheng Zhang, Hai Dong, Yan Xiao, and Shunhui Ji. 2022b. VSVC: Backdoor attack against Keyword Spotting based on Voiceprint Selection and Voice Conversion. *arXiv preprint*. ArXiv:2212.10103 [cs].

Houwei Cao, David G Cooper, Michael K Keutmann, Ruben C Gur, Ani Nenkova, and Ragini Verma. 2014. Crema-d: Crowd-sourced emotional multimodal actors dataset. *IEEE transactions on affective computing*, 5(4):377–390.

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.

Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Hai-Tao Zheng, and Maosong Sun. 2021. Openprompt: An open-source framework for prompt-learning. *arXiv preprint arXiv:2111.01998*.

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi. 2017. Neural audio synthesis of musical notes with wavenet autoencoders.

Yuan Gong, Yu-An Chung, and James Glass. 2021. Psla: Improving audio tagging with pretraining, sampling, labeling, and aggregation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2019. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244.

Asif Hanif, Maha Tufail Agro, Mohammad Areeb Qazi, and Hanan Aldarmaki. 2024a. PALM: Few-shot prompt learning for audio language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18527–18536, Miami, Florida, USA. Association for Computational Linguistics.

Asif Hanif, Fahad Shamshad, Muhammad Awais, Muzammal Naseer, Fahad Shahbaz Khan, Karthik Nandakumar, Salman Khan, and Rao Muhammad Anwer. 2024b. Baple: Backdoor attacks on medical foundational models using prompt learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 443–453. Springer.

Toni Heittola, Annamaria Mesaros, and Tuomas Virtanen. 2017. TUT Acoustic Scenes 2017, Development dataset. Technical report, Department of Signal Processing, Tampere University of Technology.

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

Zhifeng Kong, Arushi Goel, Rohan Badlani, Wei Ping, Rafael Valle, and Bryan Catanzaro. 2024. Audio flamingo: A novel audio language model with few-shot learning and dialogue abilities. *arXiv preprint arXiv:2402.01831*.

Jiahe Lan, Jie Wang, Baochen Yan, Zheng Yan, and Elisa Bertino. 2024. Flowmur: A stealthy and practical audio backdoor attack with limited knowledge. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 1646–1664. IEEE.

Siddique Latif, Moazzam Shoukat, Fahad Shamshad, Muhammad Usama, Yi Ren, Heriberto Cuayáhuitl, Wenwu Wang, Xulong Zhang, Roberto Togneri, Erik Cambria, and 1 others. 2023. Sparks of large audio models: A survey and outlook. *arXiv preprint arXiv:2308.12792*.

Shaofeng Li, Minhui Xue, Benjamin Zi Hao Zhao, Haojin Zhu, and Xinpeng Zhang. 2020. Invisible backdoor attacks on deep neural networks via steganography and regularization. *IEEE Transactions on Dependable and Secure Computing*, 18(5):2088–2105.

Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. 2022. Backdoor learning: A survey. *IEEE transactions on neural networks and learning systems*, 35(1):5–22.

Jinhua Liang, Xubo Liu, Wenwu Wang, Mark D Plumbley, Huy Phan, and Emmanouil Benetos. 2025. Acoustic prompt tuning: Empowering large language models with audition capabilities. *IEEE Transactions on Audio, Speech and Language Processing*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys*, 55(9):1–35.

Steven R Livingstone and Frank A Russo. 2018. The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. *PloS one*, 13(5):e0196391.

Karol J. Piczak. ESC: Dataset for Environmental Sound Classification. In *Proceedings of the 23rd Annual ACM Conference on Multimedia*, pages 1015–1018. ACM Press.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

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.

Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. 2020. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 11957–11965.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*.

Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. 2014. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 1041–1044.

Ashish Seth, Ramaneswaran Selvakumar, Sonal Kumar, Sreyan Ghosh, and Dinesh Manocha. 2024. Pat: Parameter-free audio-text aligner to boost zero-shot audio classification. *arXiv preprint arXiv:2410.15062*.

Cong Shi, Tianfang Zhang, Zhuohang Li, Huy Phan, Tianming Zhao, Yan Wang, Jian Liu, Bo Yuan, and Yingying Chen. 2022. Audio-domain position-independent backdoor attack via unnoticeable triggers. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, pages 583–595.

Tito Spadini. 2019. Sound events for surveillance applications.

Bob L Sturm. 2012. An analysis of the gtzan music genre dataset. In *Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies*, pages 7–12.

Yi Su, Jisheng Bai, Qisheng Xu, Kele Xu, and Yong Dou. 2025. Audio-language models for audio-centric tasks: A survey. *arXiv preprint arXiv:2501.15177*.

Mi Tian, Ajay Srinivasamurthy, Mark Sandler, and Xavier Serra. 2014. A study of instrument-wise onset detection in beijing opera percussion ensembles. In *2014 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 2159–2163. IEEE.

Alexander Turner, Dimitris Tsipras, and Aleksander Madry. 2019. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771*.

Haibin Wu, Xuanjun Chen, Yi-Cheng Lin, Kai-wei Chang, Ho-Lam Chung, Alexander H Liu, and Hung-yi Lee. 2024a. Towards audio language modeling–an overview. *arXiv preprint arXiv:2402.13236*.

Yixin Wu, Rui Wen, Michael Backes, Pascal Berrang, Mathias Humbert, Yun Shen, and Yang Zhang. 2024b. Quantifying privacy risks of prompts in visual prompt learning. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 5841–5858.

Jinwen Xin, Xixiang Lyu, and Jing Ma. 2022. Natural backdoor attacks on speech recognition models. In *International Conference on Machine Learning for Cyber Security*, pages 597–610. Springer.

Baochen Yan, Jiahe Lan, and Zheng Yan. 2024. Backdoor attacks against voice recognition systems: A survey. *ACM Computing Surveys*, 57(3):1–35.

Shaobo Zhang, Yimeng Pan, Qin Liu, Zheng Yan, Kim-Kwang Raymond Choo, and Guojun Wang. 2024. Backdoor attacks and defenses targeting multi-domain ai models: A comprehensive review. *ACM Computing Surveys*, 57(4):1–35.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022a. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16816–16825.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348.

Appendix

A Constraints on Temporal and Spectral Trigger Values

In normalized audio waveforms, amplitude values are constrained within the interval $[-1, 1]$. The perturbation budget for the temporal trigger (δ_t) is defined by $\epsilon_t \in [0, 1]$. For example, setting $\epsilon_t = 0.2$ allows the learnable δ_t values to vary within the range $[-0.2, 0.2]$. In contrast to waveform amplitudes, spectrogram coefficients vary dynamically based on audio content. To introduce perturbations in the spectral domain, we use multiplicative noise (δ_s), which scales spectrogram coefficients proportionally to their magnitudes. This enables content-aware modifications that maintain perceptual quality. Unlike additive noise, which can distort low-magnitude coefficients when the noise is too strong, or become ineffective for high-magnitude regions when too weak, multiplicative noise provides consistent, adaptive perturbations. This makes it more suitable for preserving the natural structure of the spectrogram. The perturbation budget for spectral noise is governed by $\epsilon_s \in [0, 1]$, which specifies the maximum allowable percentage change. For instance, $\epsilon_s = 0.1$ implies that δ_s lies in the range $[1 - \epsilon_s, 1 + \epsilon_s] = [0.9, 1.1]$. Values of $\delta_s < 1$ attenuate the corresponding spectrogram coefficients, while values > 1 amplify them.

B Impact of using Temporal and Spectral Triggers

Table A shows the influence of using temporal (δ_t) and spectral (δ_s) triggers independently and jointly on *Backdoor Accuracy* (BA). The attack is less effective when either trigger is used in isolation. In

contrast, combining both triggers leads to a substantial increase in backdoor effectiveness, demonstrating a strong synergistic effect.

C Impact of Perturbation Budgets

The impact of increasing the perturbation budgets (ϵ_t, ϵ_s) on the performance of the `TrojanWave` attack is illustrated in Figure A and Figure B. As the perturbation budgets increase, the attack's effectiveness significantly improves, enabling more reliable backdoor activation and higher attack success rates. However, this improvement in attack power is accompanied by a trade-off: the triggers become more perceptible, potentially raising suspicion and reducing stealth.

D Impact of Poisoning Rate

Poisoning rate refers to the percentage of training samples in which the trigger is injected and the original ground truth label is replaced with the target class label during few-shot training. The impact of increasing the poisoning rate on the performance of the `TrojanWave` attack is illustrated in Figure C. As the poisoning rate increases, the attack's success rate improves due to stronger reinforcement of the backdoor signal. However, higher poisoning rates may also increase the risk of detection and degrade the model's clean accuracy if overused, highlighting a trade-off between stealth and attack strength.

E Impact of Target Class Label

A backdoor attack requires the adversary to choose a target class label y', such that any input containing the trigger is misclassified as y', regardless of its original label. The effectiveness of the attack can vary depending on the choice of the target class, as some classes may be more susceptible due to semantic overlap with other classes or higher representation in the training data. In Figure D, we illustrate the impact of different target classes on the success rate of the `TrojanWave` attack across multiple datasets. While the attack performance remains relatively consistent across most target classes, a few outlier cases exhibit noticeably lower effectiveness, indicating that the choice of target class can occasionally influence backdoor success.

DATASETS	Temporal	Spectral	BA
Beijing-Opera	✓	✗	66.67
	✗	✓	8.330
	✓	✓	100.0
CREMA-D	✓	✗	100.0
	✗	✓	29.20
	✓	✓	100.0
ESC50-Actions	✓	✗	2.780
	✗	✓	26.39
	✓	✓	83.33
ESC50	✓	✗	58.16
	✗	✓	10.97
	✓	✓	88.78
GT-Music-Genre	✓	✗	15.64
	✗	✓	51.96
	✓	✓	92.18
NS-Instruments	✓	✗	6.790
	✗	✓	81.96
	✓	✓	99.29
RAVDESS	✓	✗	7.690
	✗	✓	98.32
	✓	✓	99.76
SESA	✓	✗	23.68
	✗	✓	28.95
	✓	✓	84.21
TUT2017	✓	✗	4.920
	✗	✓	32.95
	✓	✓	100.0
UrbanSound8K	✓	✗	60.70
	✗	✓	19.97
	✓	✓	80.35
VocalSound	✓	✗	80.15
	✗	✓	37.02
	✓	✓	97.19

Table A: **Impact of Temporal (δ_t) and Spectral (δ_s) Triggers on Backdoor Accuracy (BA) during Inference** Here ✓ indicates the corresponding trigger is used, while ✗ indicates it is not. Combining temporal and spectral triggers leads to significantly higher backdoor effectiveness than using either trigger alone.

F tSNE Plots of Embeddings of Clean and Poisoned Samples

To examine how the trigger alters feature representations, we extract embeddings from clean and poisoned audio waveforms using the audio encoder of the audio-language model. These embeddings are visualized using t-SNE plots in Figure E, which reveal that poisoned samples form distinct clusters in the embedding space, accompanied by a noticeable shift in their feature representations. This behavior contrasts with prior observations where poisoned samples were directly pulled toward the

target class cluster. Such insights can inform the design of more effective defense strategies against prompt-learning-based backdoor attacks.

G Entropy of Model Predictions on Clean and Poisoned Samples

To better understand the model's behavior under backdoor attacks, we analyze the entropy of its predictions on clean and poisoned samples. Entropy serves as a measure of confidence in the model's predictions: lower entropy indicates high confidence, while higher entropy reflects uncertainty. Distributions of entropy across all datasets is provided in Figure F. We observe that poisoned samples generally exhibit lower entropy, indicating that the model confidently misclassifies them as the adversary's target class, highlighting the strong influence of the trigger in steering model predictions. This entropy-based analysis can serve as a useful signal for detecting poisoned inputs and understanding the internal confidence dynamics of prompt-based backdoor attacks.

H Impact of Running Defense on Benign Models

Since our study does not incorporate a mechanism to verify whether a given model is benign or infected, we evaluate the impact of applying our defense method to benign models. The results show that applying the defense to a benign model has a negligible impact on clean accuracy, 76.20% before (Table 1, second column) vs. 76.14% after (Table B, second column), indicating that our method preserves performance even when no backdoor is present. This finding suggests that users can safely apply the defense even when the infection status of the model is unknown, providing a low-risk safeguard against potential backdoor threats.

I Impact of Few-Shot Training Under Clean and Defense Setups for an Equal Number of Epoch

Can additional few-shot training on clean data alone improve model performance as effectively as incorporating the defense? To investigate, we compare two scenarios: (1) training for 100 epochs on clean data, achieving 75.73% clean accuracy, and (2) training 50 epochs with the backdoor attack followed by 50 epochs applying our defense, which boosts clean accuracy to 77.31% (see Table C). These results indicate that training only on clean data may cause mild overfitting, while the defense approach provides a beneficial regularization effect, likely due to the context repulsion loss, leading to improved overall performance.

DATASETS	Benign Model	Infected Model
Beijing-Opera	97.91	93.75
CREMA-D	29.28	41.30
ESC50-Actions	97.51	95.00
ESC50	97.00	98.00
GT-Music-Genre	77.50	75.00
NS-Instruments	62.33	65.16
RAVDESS	40.12	45.62
SESA	90.48	89.52
TUT2017	82.05	81.52
UrbanSound8K	84.37	84.72
VocalSound	79.03	80.90
AVERAGE	76.14	77.31

Table B: **Impact on Clean Accuracy of Applying Defense on Benign and Infected Models** Applying the proposed defense method to either a benign or an infected model does not lead to a significant drop in *Clean Accuracy* (CA).

J Imperceptibility of Triggers: Signal-to-Noise Ratio (SNR) and Log-Spectral Distance (LSD)

We evaluate imperceptibility using two objective metrics, as reported in Table D: Signal-to-Noise Ratio (SNR) for the temporal domain and Log Spectral Distance (LSD) for the spectral domain. These metrics provide complementary perspectives—SNR measures waveform distortion, while LSD quantifies deviations in the frequency spectrum. On average, TrojanWave achieves an SNR of 29.72 dB and an LSD of 0.1198 dB across datasets. A higher SNR indicates minimal temporal distortion, while a lower LSD reflects closer spectral similarity to clean audio. Together, these results quantitatively demonstrate that the injected triggers remain largely imperceptible.

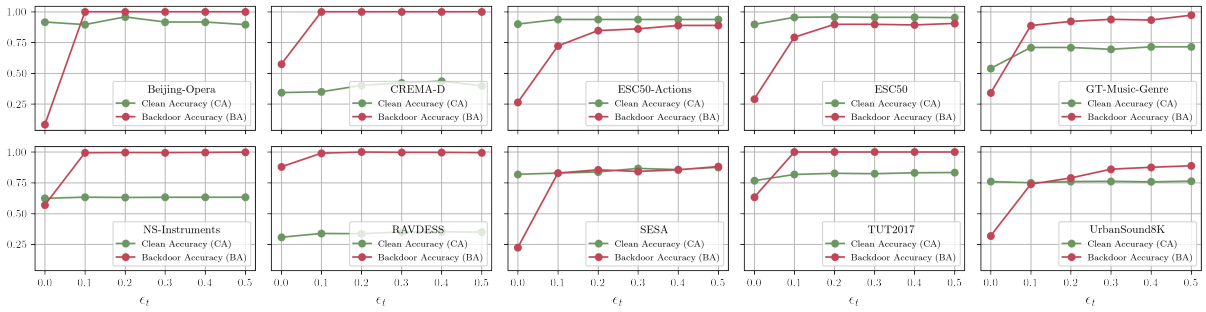


Figure A: **Impact of Temporal Trigger Perturbation Budget** (ϵ_t)

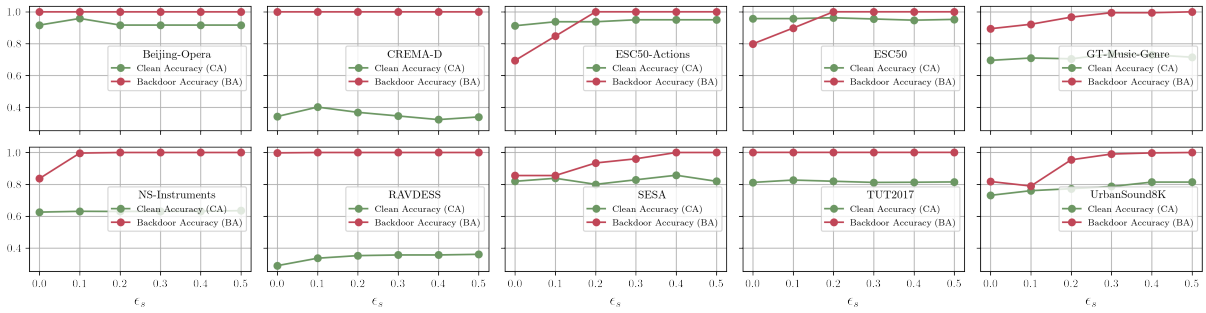


Figure B: **Impact of Spectral Trigger Perturbation Budget** (ϵ_s)

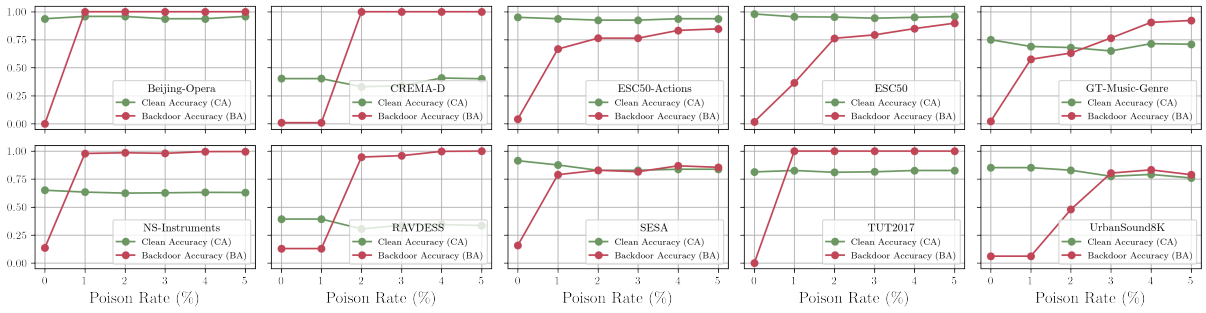


Figure C: **Impact of Poisoning Rate in Few-Shot Training Data**

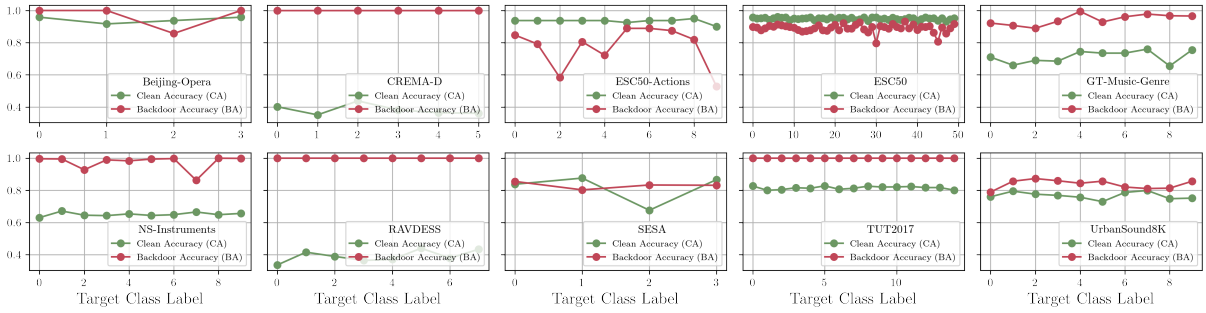


Figure D: **Impact of Target Class Label across Audio Classification Datasets**

Figure E: **tSNE Plots of Embeddings of Clean and Poisoned Samples** Each color represents a class in the dataset, with the target class highlighted in coral pink. Clean samples typically form well-defined and distinct clusters, whereas poisoned samples exhibit noticeable feature shifts. However, the overall clustering structure remains largely preserved.

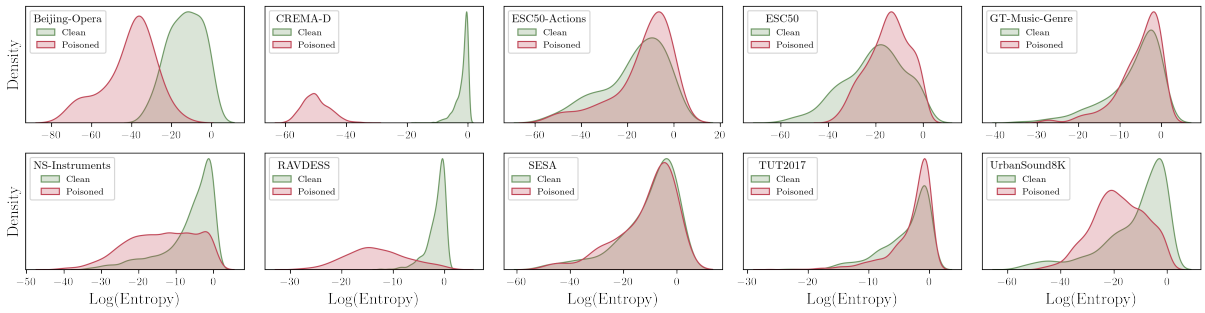


Figure F: **Entropy Distribution of Clean and Poisoned Samples**

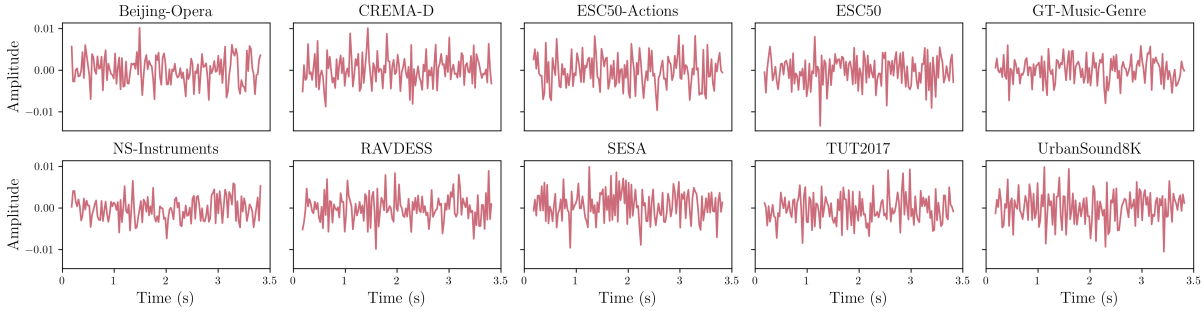


Figure G: **Optimized Temporal Triggers** (δ_t)

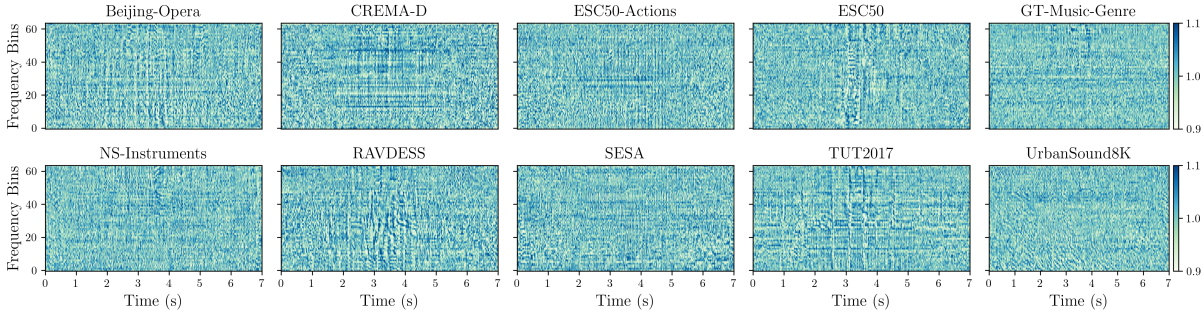


Figure H: **Optimized Spectral Triggers** (δ_s)

DATASETS	Clean Setup	Defense Setup
Beijing-Opera	97.92	93.75
CREMA-D	18.47	41.30
ESC50-Actions	97.50	95.00
ESC50	96.00	98.00
GT-Music-Genre	80.50	75.00
NS-Instruments	63.65	65.16
RAVDESS	40.94	45.62
SESA	92.38	89.52
TUT2017	83.76	81.52
UrbanSound8K	82.55	84.72
VocalSound	79.38	80.90
AVERAGE	75.73	77.31

Table C: **Impact on Clean Accuracy of Few-Shot Training Under Clean and Defense Setups for an Equal Number of Epochs** The *Clean Setup* involves running few-shot training for 100 epochs on a completely clean dataset. In contrast, the *Defense Setup* consists of two phases: the first 50 epochs of few-shot training are performed on a mixed dataset containing both clean and poisoned samples (i.e., under attack), followed by 50 epochs applying a defense method.

DATASETS	SNR (dB)	LSD (dB)
Beijing-Opera	28.83	0.1891
CREMA-D	30.05	0.1911
ESC50-Actions	30.13	0.1075
ESC50	29.90	0.1020
GT-Music-Genre	30.01	0.0986
NS-Instruments	30.35	0.0975
RAVDESS	30.66	0.0978
SESA	29.12	0.0984
TUT2017	30.58	0.0943
UrbanSound8K	29.09	0.1203
VocalSound	28.15	0.1301
AVERAGE	29.72	0.1198

Table D: **Signal-to-Noise Ratio (SNR) and Log Spectral Distance (LSD)**

DATASETS	TYPE	CLASSES
Beijing-Opera	Instrument Classification	4
NS-Instruments		10
ESC50	Sound Event Classification	50
ESC50-Actions		10
UrbanSound8K		10
CREMA-D	Emotion Recognition	6
RAVDESS		8
VocalSound	Vocal Sound Classification	6
SESA	Surveillance Sound Classification	4
TUT2017	Acoustic Scene Classification	15
GT-Music-Genre	Music Analysis	10

Table E: **Audio Classification Datasets** Audio classification datasets from various audio processing tasks have been used in this study.