ATSTA: Efficient Audio Backdoor Attack Based on Alternating Training With Limited Knowledge

Bo Wang, *Member, IEEE*, Yiming Yan, Maozhen Zhang, and Wei Wang, *Member, IEEE*

***Abstract*—As Deep Neural Networks (DNNs) have matured within audio recognition systems, the popularity of these systems has raised concerns about their security. Recently, several works revealed that these models are vulnerable to backdoor attacks. However, there are relatively few existing clean-label audio backdoor attack methods, and all of them require high poisoning rates to achieve high performance. In addition, most of the audio backdoor attacks require the attacker to have sufficient knowledge, limiting the scope of their real-world applications. To address these issues, in this paper, we propose a stealthy and efficient backdoor attack method called Alternating Training and Speech Transfer Attack (ATSTA). ATSTA enhances the attacker's knowledge by constructing a surrogate dataset and a surrogate model, then generates stealthy and efficient triggers by alternately training the surrogate model and optimizing the triggers. ATSTA has an average attack success rate of more than 99% on three victim datasets, which demonstrates the effectiveness of our approach and its robustness against State-of-the-Art (SOTA) defenses.**

***Index Terms*—AI Security, audio recognition, backdoor attack, transfer learning.**

I. Introduction

With the development of Artificial Intelligence (AI) technologies, the capabilities of audio recognition have been growing steadily. Taking Apple Siri as an example, this mode of human-computer interaction based on speech greatly facilitates user interactions [1]. However, high-performance audio recognition models require a large amount of training data and computing resources, making it difficult for users with limited resources. Therefore, many users choose to train audio recognition models by leveraging third-party services, including publicly available datasets, models, and third-party training platforms. However, recent studies have shown that third-party services may pose potential security risks [2]. Among them, backdoor attacks are one of the most typical threats [3]. Attackers insert hidden backdoors into the victim models by implanting triggers into the training audio samples of the victim models. While maintaining the model's performance on benign inputs, the audio samples containing triggers are incorrectly categorized into the attacker's desired categories [4].

Backdoor attacks first appeared in the image domain [5] and have been gradually applied to the audio domain in recent years. Based on the attack target, backdoor attacks can be categorized into model poisoning attacks and data poisoning attacks. Among them, model poisoning attacks usually occur in distributed training scenarios represented by Federated Learning [6], where the attacker injects malicious model updates during the model updating process, making the global model implanted with a backdoor. Considering that attackers usually do not have the ability to manipulate models in a single machine environment, this paper focuses on data poisoning attacks. Backdoor attacks based on data poisoning can be classified into two categories: "dirty label" and "clean label" based on label manipulation. In dirty label backdoor attacks, the attacker tampers with the training samples and modifies their labels to the target category of the attack, where the act of modifying the labels exacerbates the risk of detection by manual screening. In contrast, clean label backdoor attacks only tamper with the training samples of the target category to circumvent the checking of labels. However, most existing clean label backdoor attacks face the challenge of low attack efficiency, and more poisoned samples need to be added to the training samples or drastic modifications to the training samples to make the classifiers learn the triggers.

In addition, backdoor attacks can be divided into sample-specific and sample-agnostic triggers based on their trigger type. In the audio domain, the sample-specific trigger assigns a customized trigger to each audio sample, which enhances the stealthiness of the attack. However, this attack requires the attacker to obtain the input of the model to generate the trigger in advance, which limits the application scenarios of the attack. In contrast, the sample-agnostic trigger aims to design a generic trigger that applies to all audio samples under a certain task, which is more threatening in the real world.

In this paper, we propose a speech backdoor attack method called Alternating Training and Speech Transfer Attack (ATSTA). It aims to design an efficient audio sample-agnostic trigger that can launch an attack under limited attacker knowledge conditions (the attacker can only access the target class samples in the training dataset without accessing other samples or the

Received 2 July 2025; revised 19 November 2025; accepted 26 December 2025. Date of current version 26 January 2026. This work was supported in part by the National Natural Science Foundation of China under Grant 62372452, in part by the Application Fundamental Research Project of Liaoning Province under Grant 2025JH2/101330123, and in part by the Open Research Fund of the National Key Laboratory of Cyber Space Behavior Cognition Technology under Grant CBCT2024KF02. The associate editor coordinating the review of this article and approving it for publication was Dr. Zhangjie Fu. *(Corresponding author: Wei Wang.)*

Bo Wang, Yiming Yan, and Maozhen Zhang are with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: bowang@dlut.edu.cn; yiming@mail.dlut.edu.cn; maozhenzhang@mail.dlut.edu.cn).

Wei Wang is with the State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS) & New Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing 100190, China (e-mail: wwang@nlpr.ia.ac.cn).

Digital Object Identifier 10.1109/TASLPRO.2025.3650252

victim model), and achieve high attack performance. ATSTA aims to generate a sample-agnostic trigger that maximizes the effectiveness of the attack by alternately optimizing the surrogate model and the trigger generator on the surrogate dataset collected by the attacker. During surrogate model optimization, ATSTA incorporates backdoor samples into the training process. During the trigger optimization process, it updates the trigger generator to maximize the attack success rate on the surrogate model. In addition, we enhance trigger imperceptibility by constraining the shape of the triggers using loss functions and embedding the triggers at the location of the maximum peak of the audio samples. Subsequently, the generated triggers are migrated to the amplitude-maximizing positions of the target class samples in the attacker's possession, and the injection strength of the triggers is adjusted according to the audio strength of the samples to launch an efficient and stealthy backdoor attack on the victim model.

The main contributions of this paper are as follows:

- We propose an efficient and practical audio backdoor attack method called "Alternating Training and Speech Transfer Attack", which can be launched under the condition of limited knowledge of the attacker.
- We formalize trigger generation as an optimization problem and implement an efficient backdoor attack by alternately training surrogate models and triggers on the surrogate dataset. In a scenario with a 20% poisoning rate, ATSTA can still achieve attack success rates of over 99%.
- We design a trigger optimization strategy based on loss constraints in the trigger optimization phase and an adaptive data poisoning strategy based on maximal peak masking in the trigger injection phase to enhance the trigger imperceptibility.

II. Related Works

A. Audio Recognition System

Audio Recognition System (ARS) is an important means of human-computer interaction designed to process human audio input through algorithms or models to extract information needed by users for subsequent processing. Depending on the object to be recognized, it can be divided into *Speech Recognition System* [7], [8] and *Speaker Recognition System* [9], [10], [11]. The former extracts semantic features from the audio input and converts them into text, while the latter extracts personal voice features from the speech input to recognize the corresponding speaker from a set of predefined speakers. Early ARSs were mainly based on Gaussian Mixture Models (GMMs) [12] and Hidden Markov Models (HMMs) [13], [14]. With the application of DNNs techniques, recent ARSs have achieved advanced performance.

1) Speech Recognition System: Hinton et al. [15] first applied DNNs to the domain of speech recognition by using Feedforward Neural Networks instead of GMMs in conjunction with HMMs for acoustic modeling and achieved advanced performance in the TIMIT phoneme recognition task [16]. Graves et al. [17] investigated the application of Recurrent Neural Networks (RNNs) in the domain of speech recognition based on the advantages of RNNs in sequence data processing. In addition, inspired by ResNet [18], Vygon et al. [19] combined the ternary loss-based embedding representation with a K-Nearest Neighbors variant for classification, and designed a keyword recognition model based on ResNet, which improved the accuracy on the task of speech keyword categorization more substantially.

2) Speaker Recognition System: Reynolds et al. [20] first proposed an effective method for speaker verification based on GMMs. By using adaptation techniques to personalize the speaker model during the training process, the accuracy of the speaker verification task is significantly improved, and excellent performance is demonstrated in the NIST speaker verification task. Snyder et al. [21] proposed the X-vectors method, which aims to extract speaker embedding vectors (X-vectors) with stronger discriminative power by constructing a multilayered DNN structure to deeply encode the speech features. Chang et al. [22] proposed an end-to-end speaker verification method based on the Transformer model. This method effectively captures long-term dependencies in speech signals through the self-attention mechanism, avoiding the limitations of manual feature extraction in traditional methods.

B. Backdoor Attack

With the advancement of backdoor attacks in the visual domain, backdoor attacks against the audio recognition domain have been on the rise. The triggers of backdoor attacks can be categorized into *sample-agnostic* and *sample-specific* according to their types. Among them, the former uses a fixed audio clip as a generic trigger applicable to all audio samples under a certain task; while the latter assigns a customized trigger to each audio sample.

1) Sample-Agnostic Trigger Attacks: Liu et al. [23] achieved the first backdoor attack against the speech recognition task by adding noise and modifying labels in a digital pronunciation dataset. Zhai et al. [24] achieved the first backdoor attack against a speaker verification system by assigning different triggers to training samples in different clusters. Koffas et al. [25] realized a human imperceptible speech backdoor attack by choosing noise signals in ultrasonic frequency band as triggers. In [26] and [27], the attacker realized another covert backdoor attack by changing the natural characteristics of speech or adding sounds from the natural environment, such as bird chirps and whistles, as the trigger mode.

Despite the initial success of these sample-agnostic trigger attacks, they exhibit multiple inherent limitations. First, most traditional generic triggers suffer from labeling inconsistency, which makes poisoned samples easy to detect by manual screening. Second, these attacks require the attacker to have access to the entire victim dataset, increasing the knowledge requirement for the attacker. In addition, these attacks lack dynamism, because the triggers need to be placed at specific locations in the audio samples, increasing the difficulty of deployment for attackers in the real world.

Recently, Lan et al. [28] proposed a speech backdoor attack called FlowMur. This method presents the first dynamic backdoor attack with limited knowledge scenario of the attacker, which is dynamic, stealthy and practical. The attacker can

achieve a clean label backdoor attack by optimizing a sample-agnostic trigger on the surrogate dataset and applying it to the target class of the victim model. In addition, the triggers optimized by this method are position-independent, enabling them to physically attack the victim system in real-world scenarios and greatly improving the practicality of the method. Nevertheless, FlowMur requires at least 70% poisoning rate of the target class to achieve a high attack success rate, which increases its risk of detection.

2) Sample-Specific Trigger Attacks: Kong et al. [29] first introduced the concept of Adversarial Audio, which interferes with the model's prediction by adding small perturbations or noise to the original audio samples. By using well-designed adversarial perturbations as trigger patterns, the attacker can introduce a backdoor in the training process of the model. Ye et al. [30] proposed an optimization-based trigger generation strategy by deploying a trigger generation model to dynamically generate triggers based on samples. Cai et al. [31] exploited the sound masking effect by enhancing the pitch of samples to mask the short trigger signals with high energy in order to improve the stealthiness of the attack. In addition, they used the timbre of the audio samples as triggers, obtained the trigger timbre by additionally training a speech transition model based on StarGANv2-VC [32], and trained the backdoor model.

These sample-specific trigger backdoor attacks have the advantages of good attack effect and high stealthiness compared with the sample-agnostic trigger backdoor attacks. Such triggers based on timbre are highly stealthy [31], especially when users are not aware of the original audio, as the triggers will be much more difficult for humans to detect. However, the design of sample-specific trigger attacks requires strong knowledge of the attacker, namely the attacker needs to gain access to the entire victim dataset, which restricts their application in real-world scenarios.

III. Preliminaries

A. Threat Model

1) Attacker's Goals: In this paper, we focus on backdoor attacks based on data poisoning against ARS. The attacker aims to train a backdoored target model through stealthy manipulation of the victim dataset. The infected model must maintain normal performance on benign samples while misclassifying poisoned samples with trigger into a specific category predefined by the attacker. Stealth is critical to the attack. The trigger pattern should be imperceptible to human inspection, and the poisoning rate should be minimized to reduce detection risks. These constraints align with the practical attack scenario we investigate.

2) Attacker's Knowledge and Capability: We impose strict constraints on the attacker's knowledge, distinguishing our scenario from traditional backdoor attacks. Traditional attacks typically require adversaries to obtain the entire victim dataset, a condition that is rarely feasible in the real world and demands excessive prior knowledge. By contrast, our setting assumes the attacker only accesses samples belonging to a specific target class within the victim dataset, with no knowledge of other samples or the structure of the victim model itself. This assumption is consistent with that adopted by Lan et al. [28].

The attacker must possess public information about the victim model's learning task. For instance, if the victim model is an ARS, the attacker needs awareness that the model is specialized in recognizing audio. This requirement aligns with SOTA attacks in the image domain [33]. With such task knowledge, the attacker can select a target class from its accessible samples and collect auxiliary training data relevant to the victim's task.

Regarding capability, the attacker can only tamper with samples of the target class. To avoid detection, the attacker refrains from modifying the labels of these samples. Using the poisoned target class samples and collected auxiliary data, the attacker constructs a surrogate dataset. This dataset overlaps with the victim dataset solely in target class samples, and all other samples remain distinct. The surrogate dataset is then used to train a surrogate backdoor model, which facilitates the subsequent attack on the victim ARS. Notably, the surrogate model and the victim model may differ in architecture.

B. The Main Pipeline of Clean-Label Backdoor Attacks

For the ARS based on DNN, let $f_\theta : \mathcal{X} \to \mathcal{C}$ denote the mapping from the data space $\mathcal{X}$ to the category space $\mathcal{C}$, where the parameter θ can be learned through a training process. Suppose $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ denotes the audio dataset with N clean samples, where $x_i \in \mathcal{X}$ denotes the time series of the samples. And $y_i \in \mathcal{C} = \{0, 1, \ldots, C\}$ denotes that the dataset has C categories. The process of training this system using dataset $\mathcal{D}$ is shown in (1).

$$\theta^* = \arg\min_{\theta} \sum_{(x,y)\in\mathcal{D}} \mathcal{L}(f_\theta(x), y), \tag{1}$$

where $\mathcal{L}$ denotes the loss function, x and y correspond to the audio waveforms and their labels in dataset $\mathcal{D}$.

In the backdoor attack, the attacker retains most of the original training set to obtain the benign subset $\mathcal{B}$, and modifies the $P = \rho \cdot N$ clean samples extracted from the original training set to obtain the poisoned subset $\mathcal{P} := \{\mathcal{G}_\delta(x_i), y_t\}_{i=1}^{P}$. Where ρ denotes the poisoning rate, $\mathcal{G}_\delta(x)$ and y_t are the poisoned sample generator and target label defined by the attacker. For example, $\mathcal{G}_\delta(x) = x + \delta$ denotes the poisoned sample generation method based on additive triggers [34]. Subsequently, the benign subset is merged with the poisoned subset to create the poisoned dataset $\hat{\mathcal{D}} := \mathcal{B} \cup \mathcal{P}$, which is used to train the backdoor model. In particular, in the clean label backdoor attack, the attacker collects only $P_k = \rho \cdot M$ clean samples from the target class $k \in \mathcal{C}$ to form a subset $\mathcal{S}^k$, where M denotes the total number of clean samples of the target class. The role of the subset $\mathcal{S}^k$ is to form the poisoned subset $\mathcal{P}^k := \{\mathcal{G}_\delta(x_i), k\}_{i=1}^{P_k}$. Denoting all clean samples in the original training set $\mathcal{D}$ except for subset $\mathcal{S}^k$ as $\mathcal{B} := \mathcal{D} \setminus \mathcal{S}^k$, the poisoned training set can be denoted as $\hat{\mathcal{D}} := \mathcal{B} \cup \mathcal{P}^k$. Subsequently, the poisoned dataset $\hat{\mathcal{D}}$ is transferred to the victim to train the backdoor model, which can be expressed as (2).

$$\theta_b^* = \arg\min_{\theta} \sum_{(x,y)\in\hat{\mathcal{D}}} \mathcal{L}(f_\theta(x), y), \tag{2}$$

where x and y correspond to the audio waveforms and their labels in the poisoned dataset $\hat{\mathcal{D}}$.

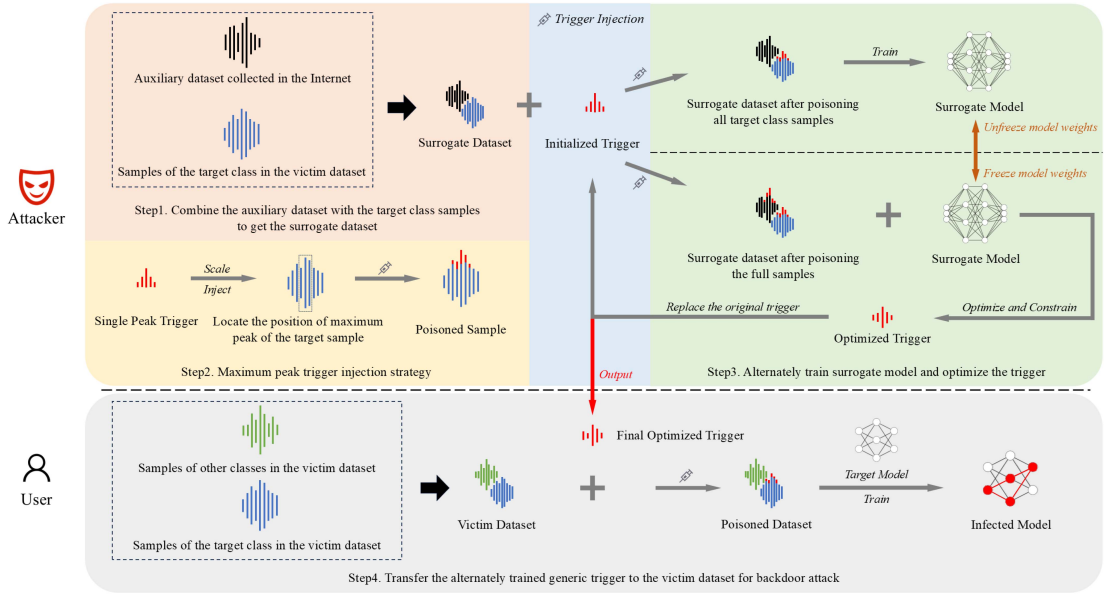

Fig. 1. Overview of ATSTA.

C. The Motivation of ATSTA

For backdoor attacks, the attacker's main goal is to design trigger patterns that combine effectiveness, stealth, and persistence. As mentioned in Section II-B, sample-specific triggers are more effective, but require strong knowledge on the part of the attacker. Transfer learning [35], [36], as a machine learning method, aims to improve the learning of new tasks by migrating knowledge from relevant tasks that have already been learned, in order to solve the data dependency problem of deep learning [37]. We argue that triggers for backdoor attacks designed for the same learning task are transferable. Therefore, we propose to train backdoor models by training on a large number of related task datasets and transferring the trigger to the victim dataset for backdoor attacks. In addition, a high-performance classification task based on DNN is achieved by optimizing the loss function, which allows the model to learn the features of audio samples by updating the parameters in each iteration to achieve accurate classification of the samples. Inspired by this optimization idea, we believe that optimization through multiple iterations helps the model learn data features better and improves generalization ability and stability. So we formalize trigger generation as an optimization problem. We maximize the poisoning effect of the trigger by alternating between training the model and optimizing the trigger during the process of training the surrogate model. We refer to the proposed method as ATSTA, which obtains an efficient speech trigger by alternating training and transfers it to the victim model for backdoor attack, the technical details of which are described in Section IV.

IV. Methodology

A. Overview of Our Method

Our proposed ATSTA takes into account the lack of knowledge of the attacker in real-world scenarios. As shown in Fig. 1, the ATSTA consists of four main parts: "Surrogate Dataset Acquisition", "Trigger Design", "Alternating Training" and "Transfer Attack". Among them, the "Surrogate Dataset Acquisition" part is used to enhance the attacker's knowledge and train the trigger based on it. In the "Trigger Design" section, we propose a maximum peak based trigger injection strategy and a trigger optimization strategy based on loss constraints to achieve covert trigger injection. In the "Alternating Training" section, we inject the initialized trigger into the surrogate model and alternate training the model while optimizing the trigger. In this process, we restrict the shape of the trigger to enhance the stealthiness of the attack. Finally, in the "Transfer Attack" section, we use the final optimized trigger to attack the victim model in order to realize the backdoor attack under the scenario of attacker's lack of knowledge.

B. Surrogate Dataset Acquisition

According to the assumption on the attacker's knowledge in Section III-A2, the adversary can only access a certain class in the victim's dataset D. Therefore, the attacker may designate this class as the target class of the backdoor attack, and denote the dataset corresponding to this target class as D_t. To solve the problem of insufficient knowledge, the adversary will collect samples related to the learning task of the victim model from the Internet and other sources to build the auxiliary dataset D_{aux}. For example, if the task of the victim model is the recognition of keywords in English speech, the adversary can collect samples of commonly-used English audio samples, such as "yes", "no" and other keywords. Subsequently, the adversary combines the auxiliary dataset D_{aux} with the target class dataset D_t to obtain the surrogate dataset $D_{sur} := D_{aux} \cup D_t$, on which the surrogate model $f_{\theta_{sur}}(\cdot)$ is trained and the triggers are optimized.

C. Trigger Design

In this section, we present the trigger design scheme, which consists of two parts, trigger injection and trigger optimization, for maximizing the stealthiness of the trigger. In the trigger injection part, we propose a Maximum Peak Masking Trigger Injection Strategy to enhance the stealthiness of the triggers. In the trigger optimization part, we design a trigger optimization strategy based on loss constraints, which restricts the shape of the trigger by multiple loss functions to ensure its imperceptibility to the human.

1) Maximum Peak Masking Trigger Injection Strategy: Ma et al. [38] proposed an image hiding method based on transform domain, but it is difficult to directly apply to the audio domain. Micheyl et al. [39] and Zwicker et al. [40] showed that the human auditory system has a limited ability to distinguish overlapping sounds, which means that in complex mixtures of sounds, subtle transformations can be masked by other sounds. Based on this, we propose the Maximum Waveform Masking Trigger Injection Strategy, which enhances the stealthiness of backdoor attacks by covertly implanting triggers using sound masking.

As a mechanical wave, sound waves have peaks and valleys. Among them, the signal energy at the peak is more concentrated, which provides a natural condition for sound masking. In addition, the envelope shape of the speech signal shows a peaked trend. Therefore, we set the initial shape of the trigger to a single-peak shape, and align the peak of the trigger with the peak of the largest amplitude in the speech samples during the trigger injection process to achieve the most stealthy attack. Specifically, to find these locations, we traverse the full audio samples to identify the peak with the largest amplitude in each sample. Consequently, the optimal injection location T is determined by (3).

$$T = \arg\max_{i} \left(\sum_{j=i-\frac{L}{4}}^{i+\frac{L}{4}} |x_j| \right), i \in \left[\frac{L}{4}, n - \frac{L}{4} \right], \tag{3}$$

where $|x_j|$ denotes the amplitude of the j^{th} element in the speech sequence, L is the length of the trigger, and n is the length of the speech signal. We calculate the cumulative amplitude within a window of length $\frac{L}{2}$ centered at index i to avoid selecting locations with only a single anomalous peak.

Subsequently, we inject the trigger into the audio sample for data poisoning. In the process, the attacker aligns the midpoint position τ of the trigger sequence with the maximum amplitude position T, and appends the trigger δ to the audio sample. It is important to note that the imperceptibility of the trigger pattern varies from one host sample to another. For example, it is difficult for humans to perceive noise in noisy environments. But in quiet environments, a slight noise may also be harsh. Therefore, to further enhance the imperceptibility of the trigger, we use the adaptive data poisoning method based on SNR [41] designed by Lan et al. [28], to assign a different trigger scaling to each audio sample based on the SNR specified by the attacker. The formula of SNR is shown in (4).

$$\text{SNR} = 10 \lg \frac{\|x\|_2^2}{\|\gamma \cdot \delta\|_2^2}, \tag{4}$$

where $\|x\|_2^2$ is the square of the Euclidean norm of the audio sample x, $\|\gamma \cdot \delta\|_2^2$ is the square of the Euclidean norm of the scaled trigger $\gamma \cdot \delta$, and γ is the scaling factor of the trigger δ with respect to the sample x. The formula can be expressed as (5).

$$\gamma = 10^{-\frac{\text{SNR}}{20}} \frac{\|x\|_2}{\|\delta\|_2}. \tag{5}$$

Based on the above analysis, the process of trigger injection can be represented as shown in (6).

$$x_p = \mathcal{G}_\delta^\gamma(x) = x + \gamma \cdot \delta(\tau = T), \tag{6}$$

where x_p denotes the poisoned sample after trigger injection, $\mathcal{G}_\delta^\gamma(x)$ is the poisoned sample generation function. $\delta(\tau = T)$ denotes the alignment of the midpoint position τ of trigger δ with the maximum amplitude position T under the audio samples.

2) Trigger Optimization Strategies Based on Loss Constraints: The purpose of this section is to generate a sample generic trigger δ^*. When δ^* is injected into any sample x with the same learning task as the victim model, it causes the infected model $f_{\theta'}(\cdot)$ to classify the poisoned samples x_p as target label y_t.

$$f_{\theta'}(x_p) = y_t, \tag{7}$$

where x_p is the poisoned sample obtained in (6).

The trigger δ^* can be implemented by minimizing the following objective function.

$$\begin{aligned} \delta^* &= \arg\min_{\delta} \sum_{(x_p, y) \in \hat{D}} \mathcal{L}\left(f_{\theta'}(x_p), y_t\right) \\ &= \arg\min_{\delta} \sum_{(x,y) \in D} \mathcal{L}\left(f_{\theta'}\left(\mathcal{G}_\delta^\gamma(x)\right), y_t\right), \end{aligned} \tag{8}$$

where $\mathcal{L}(\cdot)$ denotes the loss function, which measures the difference between the model's prediction of the poisoned sample x_p and the target label y_t. D denotes the original dataset and $\hat{D}$ denotes the poisoned dataset. Note that $\hat{D}$ is obtained by poisoning the entire sample in D.

To enhance stealthiness, we want the difference between poisoned samples and clean samples to be as small as possible. Li et al. [42] enhanced the stealthiness of the attack by introducing a perceptual loss based on feature extraction. We bring the clean samples closer to the poisoned samples through the Mean Square Error (MSE) loss [43]. The MSE loss constraint can be formalized as shown in (9).

$$\begin{aligned} \mathcal{L}_{MSE}(x, x_p) &= \mathcal{L}_{MSE}\left(x, \mathcal{G}_\delta^\gamma(x)\right) \\ &= \frac{1}{N} \sum_{n=1}^{N} \left(x^n - \mathcal{G}_\delta^\gamma(x^n)\right)^2, \end{aligned} \tag{9}$$

where $\mathcal{L}_{MSE}(\cdot)$ denotes the MSE loss, N represents the batch size, and n is the sample index within each batch. Although

Algorithm 1: Alternating Training.

Input : Auxiliary dataset D_{aux}, target dataset D_t, surrogate dataset D_{sur}, target label y_t, surrogate model $f_{\theta_{sur}}(\cdot)$, poisoned sample generation function $\mathcal{G}_{\delta}^{\gamma}(\cdot)$, hyperparameters SNR, λ_1, λ_2, ϵ

Output: Final Optimized Trigger δ^*

Initialize Single Peak Trigger δ_0
for *number of epochs* **do**
  for $(x_b, y_b) \in D_{aux}$ *and* $(x_t, y_t) \in D_t$ **do**
    $x_t' \leftarrow \mathcal{G}_{\delta_i}^{\gamma}(x_t)$; // Get poisoned sample from Equation 6
    $L_b = \mathcal{L}(f_{\theta_{sur}}(x_b), y_b)$; // Get benign sample loss
    $L_p = \mathcal{L}(f_{\theta_{sur}}(x_t'), y_t)$; // Get poisoned sample loss
    $\theta_{sur}' \leftarrow \arg\min_{\theta_{sur}} \sum [L_b + L_p]$; // Get infected model
  end
  Freeze the weights of the infected surrogate model $f_{\theta_{sur}'}(\cdot)$
  for $(x, y) \in D_{sur}$ **do**
    $x' \leftarrow \mathcal{G}_{\delta_i}^{\gamma}(x)$; // Get poisoned sample from Equation 6
    $L_p = \mathcal{L}(f_{\theta_{sur}'}(x'), y_t)$; // Get poisoned sample loss
    $L_{MSE} = \mathcal{L}_{MSE}(x, x')$; // Get MSE loss from Equation 9
    $L_{TV} = \mathcal{L}_{TV}(\delta_i)$; // Get TV loss from Equation 10
    $L_{total} = L_p + \lambda_1 \cdot L_{MSE} + \lambda_2 \cdot L_{TV}$; // Calculate the total loss from Equation 11
    $\delta_{i+1} \leftarrow \arg\min_{\delta} \sum L_{total}$; // Update trigger from Equation 11
    $\delta_{i+1} = clip(\delta_{i+1}, [-\epsilon, \epsilon])$; // Constraint trigger
  end
  Unfreeze the weights of the infected surrogate model $f_{\theta_{sur}'}(\cdot)$
end

the MSE loss ensures a certain degree of alignment between poisoned samples and clean samples, it exhibits inherent sensitivity to outliers. This characteristic introduces significant noise into the generated triggers. To address this issue, we adopt the Total Variation (TV) loss [44] to smooth the triggers and mitigate potential noise in the poisoned samples. Specifically, the constraint imposed by the TV loss is formalized in (10).

$$\mathcal{L}_{TV}(\delta) = \frac{1}{N} \sum_{n=1}^{N} \sum_{i=1}^{L-1} \left| \delta_{i+1}^{n} - \delta_{i}^{n} \right|, \tag{10}$$

where $\mathcal{L}_{TV}(\cdot)$ is the TV loss, L denotes the length of a single trigger sequence and i is the element index of each trigger.

At the same time, speech triggers obtained based on the optimization method tend to generate loud noise, therefore the amplitude of the audio signal needs to be limited. Ultimately, we find the best trigger δ^* by solving the following optimization problem.

$$\delta^* = \arg\min_{\delta} \sum_{(x,y)\in D} [\mathcal{L}\left(f_{\theta'}\left(\mathcal{G}_{\delta}^{\gamma}(x)\right), y_t\right) + \lambda_1 \cdot \mathcal{L}_{MSE}\left(x, x_p\right) + \lambda_2 \cdot \mathcal{L}_{TV}\left(\delta\right)], \delta \in [-\epsilon, \epsilon], \tag{11}$$

where λ_1 and λ_2 are hyperparameters used to control the weights of the loss function, and $\delta \in [-\epsilon, \epsilon]$ is used to limit the amplitudes of δ.

D. Alternating Training

Due to the interdependence between the infected model $f_{\theta'}(\cdot)$ and the trigger δ, the infected model needs to inject the given trigger δ into the samples before training. But the optimization of the trigger δ needs to be performed on the given infected model $f_{\theta'}(\cdot)$, so the solution of (11) is very difficult. Shi et al. [45] solved this problem by joint optimization of the infected model and the trigger. However, this approach requires the attacker to have fairly strong knowledge, namely access to the victim model $f_{\theta'}(\cdot)$ and the victim dataset D. At this time, supplementing the attacker's knowledge with the surrogate dataset D_{sur}, along with alternately training the victim model and optimizing the trigger becomes a feasible approach to solving the above problem.

Algorithm 1 is used to solve (11). Specifically, we first initialize a quadratic-shaped single-peak trigger δ_0, and divide each training epoch of ATSTA into two parts. In the first half of the i^{th} training cycle, δ_i is injected into the target class samples in the surrogate dataset D_{sur} to simulate the poisoning process of a clean label backdoor attack, on which an infected surrogate model $f_{\theta_{sur}'}(\cdot)$ is trained. In the second half, we first freeze the weights of the surrogate model $f_{\theta_{sur}'}(\cdot)$. Subsequently, we inject the trigger δ_i into all the samples in the surrogate model D_{sur} to improve the generalization of the triggers, then optimize the trigger based on the loss function. Among them, the Cross-Entropy loss [46] is used to make the prediction results of the poisoned samples on the model $f_{\theta_{sur}'}(\cdot)$ to establish a connection with the target class label y_t. The MSE loss and the TV loss are used to enhance the imperceptibility of the trigger. Combining these three loss functions, the trigger is optimized using the Adam optimizer [47]. Subsequently, the amplitude of the trigger is constrained to be in the range $[-\epsilon, \epsilon]$, using the constraint function $clip(\cdot)$ to obtain the trigger δ_{i+1} for this epoch, and to further improve the imperceptibility. Finally, the

weights of the surrogate model $f_{\theta'_{sur}}(\cdot)$ are unfrozen, and the optimized trigger δ_{i+1} is used to replace the previous trigger δ_i to start a new round of training.

E. Transfer Attack

The last part of ATSTA is the transfer attack. The attacker injects the final optimized trigger δ^* into the target class dataset D_t in their possession to obtain the poisoned target class dataset D'_t, which will be delivered to the dataset publisher. The dataset publisher combines D'_t with other classes of victim dataset $D_{/t}$ to obtain the poisoned dataset $D' = D'_t \cup D_{/t}$, and delivers it to the victim to train the model. Any DNN model trained with this poisoned dataset D' will be embedded with a backdoor, and its learning process can be represented as shown in (12).

$$\theta' = \arg\min_{\theta} \sum_{(x,y)\in D'} \mathcal{L}(f_\theta(x), y), \tag{12}$$

where $f_\theta(\cdot)$ denotes the victim model and $f_{\theta'}(\cdot)$ is the infected victim model.

The infected model with a backdoor is not noticeable to the user as it behaves normally on benign samples. However, it will categorize any poisoned samples with triggers as the attacker-specified target label y_t.

V. Experiments

A. Experimental Setup

1) Datasets: We evaluate the ATSTA method using three of the most classical speech datasets: Google Speech Command Dataset (SCD) [48], Football Keywords Dataset (FKD) [49] and AudioMNIST [50]. Specifically, the three datasets are described below:

Speech Command Dataset: SCD is one of the most classical English keyword recognition datasets, containing 35 common spoken English commands. It contains 105,829 audio samples, each with a sampling rate of 16 kHz and a duration of about 1 s. In our experiments, we select 10 categories out of all 35 categories as the victim dataset D and use the remaining 25 categories for the auxiliary dataset D_{aux}.

Football Keywords Dataset: FKD is a publicly available Persian keyword recognition dataset, containing 18 spoken Persian commands related to football matches. It contains 30,259 audio samples, each with a sampling rate of 16 kHz and a duration of 1.25 seconds. In our experiments, we select 8 categories out of all 18 categories as the victim dataset D and use the remaining 10 categories for the auxiliary dataset D_{aux}.

AudioMNIST: AudioMNIST is a publicly available English digitized speech dataset containing 10 digits from 0 to 9, provided by 60 speakers. It contains 30,000 audio samples, each with a sampling rate of 16 kHz. In addition, the dataset covers a wide range of features including speaker ID, gender, accent, age, and so on. In our experiments, we choose this dataset to validate the attack effect of ATSTA in speaker recognition system by selecting 25 categories out of all 60 speaker categories as the victim dataset D, and use the remaining 35 categories for the auxiliary dataset D_{aux}.

TABLE I
The Recognition Accuracy of the Six Models on the Three Datasets (%)

Dataset \ Model	SmallCNN	LargeCNN	ResNet	RNN	LSTM	KWT
SCD	96.65	97.80	97.04	92.84	96.90	87.29
FKD	96.37	99.04	98.58	87.88	97.79	87.46
AudioMNIST	98.88	98.56	98.50	97.56	97.12	85.44

To maintain the consistency of the experimental setup, we adjusted all audio clips to 1 s and equalized their loudness.

2) Models: In ATSTA, we assume that the attacker has no knowledge of the victim model. To evaluate the effectiveness against different DNNs, we conduct experiments on six models: two Convolutional Neural Networks (CNNs), ResNet [18], RNN [51], LSTM [52], and KWT [53]. Specifically, the two CNNs are denoted as SmallCNN and LargeCNN. The former contains three convolutional layers and two fully connected layers, while the latter contains five convolutional layers and three fully connected layers. As an advanced speech recognition model, KWT adopts the architecture of Transformer. Detailed network architectures follow the configurations provided in [28] and [31]. We train these models on the three datasets using the target class training data held by the attacker and record their recognition accuracies in the absence of attacks as shown in Table I.

3) Baseline Methods and Defense Mechanisms: Considering the relative rarity of backdoor attacks in scenarios with limited attacker knowledge, we choose Natural Backdoor Attack (NBA) [27] and FlowMur as the baseline method. Specifically, FlowMur is the current SOTA result. The experiments of Xin et al. [27] show that using whistles as the trigger of NBA is relatively effective, so we adopt whistles as the trigger of NBA in our experiments. For defense methods, most backdoor defense methods in the image domain cannot be directly migrated to the audio domain. Therefore, we choose three methods, Fine-Pruning [54], STRIP [55], and Beatrix [56] to verify the robustness of ATSTA, because they are proven to be effective in the audio domain.

4) Evaluation Metrics: The experimental evaluation metrics include Benign Accuracy (BA) and Attack Success Rate (ASR), which measure the performance of the backdoor attack under normal categorization tasks and specific trigger conditions, respectively. Specifically, BA measures the proportion of benign samples which are categorized with correct labels, while ASR indicates the proportion of poisoned samples which are predicted to be the target label specified by the attacker. In addition, we invited 20 volunteers to evaluate the stealthiness of ATSTA. Considering that there are more benign samples than poisoned samples in real scenarios, we randomly selected 35 benign samples and 5 poisoned samples at a time to be given to the volunteers for judgment, and repeated the test 3 times for everyone. We define the proportion of poisoned samples judged as malicious by volunteers as Exposure Rate (ER). The lower the ER, the more stealthy the attack is.

5) Default Training Setup: We choose the Mel Frequency Cepstrum Coefficient (MFCC) as the input feature and set the

TABLE II
IMPACT OF SURROGATE-TARGET MODEL CATEGORIES ON BA/ASR ON THREE DATASETS (%)

Dataset	Tar \ Sur	SmallCNN	LargeCNN	ResNet	RNN	LSTM	KWT
SCD	SmallCNN	96.05/99.95	97.69/99.96	97.65/99.31	94.04/98.7	97.17/99.15	86.28/99.62
	LargeCNN	96.87/99.68	97.62/100	97.61/99.85	95.06/99.05	96.95/99.95	86.23/99.26
	ResNet	96.5/99.88	96.69/99.95	97.4/99.52	94.86/98.13	97.01/99.91	85.78/99.66
	RNN	96.97/88.69	96.89/89.99	97.44/77.05	94.16/90.49	96.61/86.83	86.32/86.64
	LSTM	96.81/99.91	97.6/99.73	97.57/97.09	94.94/98.42	97.15/99.8	85.76/99.25
	KWT	96.71/99.37	97.79/90.15	97.94/97.19	95.82/95.01	96.62/99.53	87.6/99.13
FKD	SmallCNN	96.95/95.32	97.77/98.44	97.86/95.34	93.78/76.35	97.33/97.9	87.01/99.05
	LargeCNN	97.61/99.83	98.24/99.42	97.94/95.88	93.26/86.97	97.77/95.04	84.04/96.75
	ResNet	96.70/88.24	98.87/90.56	98.44/95.39	93.32/98.41	97.68/95.66	86.04/98.84
	RNN	97.11/81.42	97.45/86.34	98.96/82.49	93.74/81.05	97.51/82.61	74.58/91.69
	LSTM	97.43/96.53	98.13/90.11	98.21/95.43	94.69/80.06	98.34/92.54	85.27/98.98
	KWT	97.50/99.30	98.01/90.98	98.94/90.09	95.55/88.75	98.22/95.29	84.57/97.64
AudioMNIST	SmallCNN	99.3/100	98.1/99.76	99.22/99.03	98.6/99.98	98.10/99.27	85.01/95.03
	LargeCNN	98.75/99.98	99.36/100	99.12/99.70	97.55/100	98.61/99.84	85.05/95.39
	ResNet	98.3/99.45	99.31/99.98	99.08/98.94	98.50/99.90	97.97/99.01	86.08/95.08
	RNN	98.31/79.12	97.95/89.43	99.42/83.59	97.04/80.88	97.96/85.21	88.59/78.73
	LSTM	98.65/99.90	99.17/98.97	98.72/98.7	98.23/100	97.67/99.17	84.26/91.96
	KWT	98.65/98.73	98.04/99.06	98.99/98.58	98.56/99.1	98.05/97.95	90.32/98.25

sampling rate to 16 kHz. For SCD and AudioMNIST, we partition them according to 64% for training, 16% for validation, and 20% for testing. For FKD, the training and validation sets are set to maintain a 4:1 ratio, while the samples in the testing set are kept the same as the original dataset. In addition, the size of D_{aux} should be larger than D because the attacker needs to collect a large amount of auxiliary data to enhance the attacker's knowledge. During alternate training, we use the Adam optimizer to alternately train the surrogate model with the optimized trigger, with a total of 100 training epochs. Specifically, we set the learning rate to 0.008 for the KWT model and 0.001 for the other models, and 0.01 for the triggers. By default, we use the SmallCNN as the surrogate model architecture and use LargeCNN as the target model architecture. For other parameters: We set the trigger duration to 0.3 seconds for SCD, 0.4 seconds for AudioMNIST, and 0.6 seconds for FKD. The target class poisoning rate is set to 20%, SNR is set to 20 dB, with λ_1 and λ_2 set to 0.001 and 0.0001, respectively. These parameters are further analyzed in Sections V-B and V-D.

B. Effectiveness of Backdoor Attacks

1) Impact of Surrogate-Target Model Class on Attack Performance: First, we combine surrogate models with target models pairwise to verify the effect of model selection on attack performance, and the specific results are presented in Table II.

As indicated in Table II, ATSTA maintains competitive performance across the three datasets for various surrogate-target model combinations. Regarding BA, its performance on all three datasets is nearly comparable to that on benign datasets. For ASR, 36 combinations are considered for all the three datasets. Among these, 23 combinations on SCD and 19 combinations on AudioMNIST achieve ASR exceeding 99%, while 29 combinations on both datasets reach ASR higher than 95%. For FKD, 20 combinations yield ASR above 95% and 26 combinations achieve ASR over 90%. The relatively lower ASR on FKD is attributed to the high inherent noise of this dataset. The noise overlaps with the generated trigger, which impairs the model's ability to recognize valid triggers.

We further observe distinct impacts of model selection on attack performance. Utilizing RNN as either the surrogate model or the target model consistently results in relatively low ASR. In addition, combinations based on KWT also lead to partial degradation of ASR. This discrepancy originates from structural differences between these models and other participating models. Different from the two CNNs and ResNet, which rely on convolutional and pooling layers as basic units, RNN adopts recurrent neurons and KWT incorporates an attention mechanism. These structural divergences affect the models' sensitivity to the trigger. In contrast, combinations involving LSTM exhibit only slight ASR reduction on FKD. This phenomenon is likely due to the gating mechanism of LSTM, which enables more effective capture of trigger sequences and thus delivers more robust attack performance compared to RNN and KWT.

Notably, even in the least favorable scenarios, specifically when RNN serves as either the surrogate or target model, ATSTA still achieves minimum ASR values of 77.05% on SCD, 76.35% on FKD, and 78.73% on AudioMNIST. These results clarify the impact of model selection on attack performance: Networks with recurrent units or attention mechanisms tend to reduce attack performance when employed as surrogate or target models. Nevertheless, the consistent performance of ATSTA across diverse model combinations underscores its effectiveness in practical attack scenarios.

2) Impact of Victim Dataset to Auxiliary Dataset Ratio on Attack Performance: To generate the most effective trigger for the victim model, we study the impact of data balance on attack performance by regulating the partition ratio between the victim dataset D_t and the auxiliary dataset D_{aux}. The experimental results are presented in Table III.

As indicated in Table III, ATSTA exhibits a consistent variation trend across the three datasets under different partition ratios of D_t and D_{aux}. As the D_t/D_{aux} ratio increases, the BA

TABLE III
IMPACT OF THE PARTITION BETWEEN VICTIM DATASET AND AUXILIARY DATASET ON THE ATTACK PERFORMANCE ON THREE DATASETS (%)

SCD				FKD				AudioMNIST			
D_t	D_{aux}	BA	ASR	D_t	D_{aux}	BA	ASR	D_t	D_{aux}	BA	ASR
5	30	97.72	99.59	4	14	98.58	99.11	5	55	98.4	97.32
10	25	96.38	99.32	6	12	98.15	98.93	15	45	97.65	98.51
15	20	96.87	99.68	8	10	97.61	99.83	25	35	98.75	99.98
20	15	94.98	99.04	10	8	98.38	94.54	35	25	95.61	91.62
25	10	94.01	99.32	12	6	97.63	95.05	45	15	92.6	85.04
30	5	92.94	98.84	14	4	96.75	93.35	55	5	92.76	76.67

of ATSTA decreases progressively, while the ASR demonstrates a trend of initial increase followed by decrease. This variation is most pronounced on AudioMNIST.

This phenomenon can be attributed to the following mechanisms. A higher D_t/D_{aux} ratio implies diminished coverage of D_t by D_{aux}. Such limited coverage may cause the generated triggers to interfere with the model's primary task, thereby reducing BA. Regarding the variation of ASR, when D_{aux} is significantly larger than D_t, ATSTA incorporates an excessive number of categories irrelevant to the victim dataset during trigger optimization, which reduces the specificity of the resulting triggers. Conversely, when D_{aux} is considerably smaller than D_t, the trigger fails to cover all categories of D_t in the optimization process. Both scenarios lead to diminished attack effectiveness.

Nevertheless, ATSTA maintains stability and effectiveness on both SCD and FKD. When D_{aux} is slightly larger than D_t, ATSTA achieves the optimal attack performance across all three datasets while ensuring the accuracy of the model's primary task.

3) Impact of Poisoning Rate on Attack Performance: In order to study the effect of the target category poisoning rate on the attack performance, we investigate the effect of increasing the poisoning rate from 10% to 100% on the ATSTA with a step size of 10%. In particular, considering the high efficiency of the ATSTA attack, we also investigate the effect on the attack performance at 5% poisoning rate, and the experimental results are shown in Fig. 2.

As shown in Fig. 2, ATSTA can still achieve over 90% ASR on all three datasets at 5% poisoning rate, while it can achieve over 98% ASR on all three datasets when the poisoning rate reaches 20%. In contrast, FlowMur requires 70% poisoning rate of the target class to achieve over 90% ASR on all three datasets, and the NBA is unable to break 80% ASR even with a 100% poisoning rate. Compared with FlowMur, we believe the reason for this is that the triggers obtained by ATSTA through alternate training are more closely linked to the attacker's target class data and more closely fit the learning task of the victim model. In contrast, NBA uses randomly specified triggers, and it is unable to establish a strong enough connection between the trigger and the target class data. Among them, NBA has a relatively high ASR on AudioMNIST due to the fact that AudioMNIST is simpler and less than 1 s in length, and we get samples of 1 s in length by complementing each sample with 0. The trigger may be added at amplitude 0, which leads to an elevated ASR. In addition, when the target class poisoning rate is close to 100%, the BA of the three methods decreases sharply.

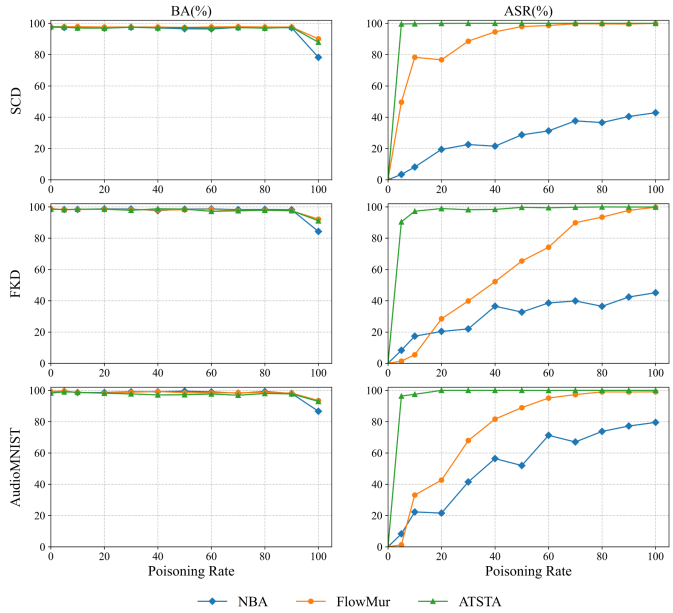

Fig. 2. Impact of target class poisoning rates on the effectiveness of NBA, FlowMur and ATSTA.

The reason for this is that when the target class poisoning rate is 100%, all target class samples contain triggers, which prevents the model from learning its features. Among them, the BA of ATSTA and FlowMur decrease less, which may be due to the fact that both of them impose restrictions on the location of the trigger attachment, thus weakening the trigger's perturbation on the features of the benign samples.

4) Impact of Trigger Duration: In order to investigate the effect of trigger duration on attack performance, we increase the trigger duration from 0.1 seconds to 1 s in steps of 0.1 seconds on the three datasets, and the experimental results are shown in Fig. 3.

As can be seen from Fig. 3, the trigger duration does not lead to a significant change in BA. This is because the model can learn the intrinsic characteristics of the target class samples from the benign samples. Meanwhile, on the three datasets, ASR shows an increasing trend with the increase of trigger duration. And the gap between ATSTA and the other two methods gradually increases as the trigger duration increases.

This is because we set up a scenario with a 20% poisoning rate, and it is difficult for the victim to establish the link between the trigger and the target label through the poisoned samples of FlowMur and NBA. On SCD and AudioMNIST, ATSTA

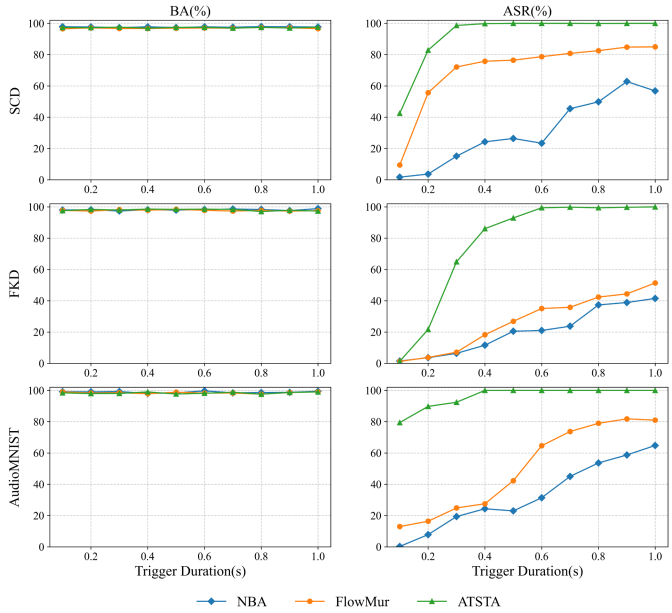

Fig. 3. Impact of trigger duration on the effectiveness of NBA, FlowMur and ATSTA.

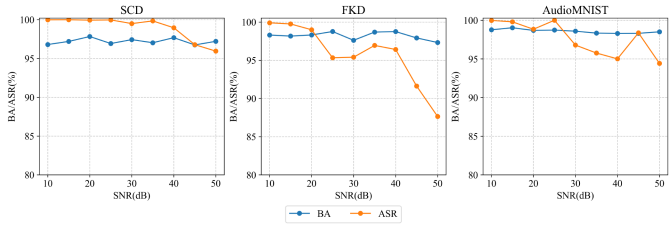

Fig. 4. Impact of SNR on the effectiveness of ATSTA.

achieves high ASR even at lower trigger durations. This is due to the higher similarity between the intrinsic features of the triggers we obtained through alternate training and the features of the target category samples. And the poisoned model can easily categorize samples with triggers into the target class. In addition, we found that on FKD, ATSTA requires a trigger duration of 0.6 seconds or more to achieve a sufficiently high ASR, possibly because the samples in FKD are recorded in a noisy environment with high internal noise, and more distinctive triggers are needed to establish a link with the target label.

5) Impact of SNR: Since SNR-based trigger scaling is used in ATSTA, we verified the effect of SNR between benign samples and trigger audio on the performance of ATSTA attacks. The results of the experiments are shown in Fig. 4 by increasing the SNR from 10 dB to 50 dB in steps of 5 dB on the three datasets.

From Fig. 4, it can be seen that the change of SNR does not significantly affect BA. The reason is that the model can learn its intrinsic features from benign samples. Meanwhile, for SCD and AudioMNIST, a decrease in ASR can be observed when the SNR exceeds 25 dB. This is because the injected triggers become less prominent relative to the benign signal as SNR increases, making it difficult for the model to capture the features of the triggers. Nonetheless, the ASR can still exceed 93% on both datasets, which demonstrates the robustness of ATSTA to

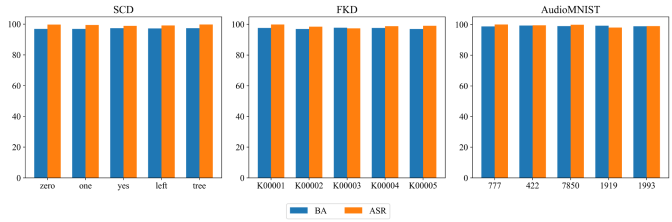

Fig. 5. Impact of target categories on the effectiveness of ATSTA.

TABLE IV
EXPOSURE RATE OF POISONED SAMPLES UNDER NBA, FLOWMUR AND ATSTA (%)

Benign	NBA	FlowMur	ATSTA
0	81.33	18.67	9

changes in SNR. Moreover, a significant decrease in ASR on FKD can be observed when the SNR exceeds 20 dB and 40 dB. The possible reason for this is that the samples of FKD were recorded in a noisy environment, while the samples of SCD and AudioMNIST were collected in a quiet environment. Therefore, as the SNR increases, the injected triggers in the dataset of FKD are more perturbed by noise, which leads to a relatively low ASR.

6) Impact of Target Categories: In order to verify the effectiveness of our ATSTA's attack under different target categories, we select five categories on each of the three datasets as the attacker's target categories. As shown in Fig. 5, ATSTA has similar attack effectiveness on different target categories on each dataset. Specifically, the BA difference across different target categories on the same dataset is within 1%. Additionally, the ASR is greater than 97% on a total of 15 target categories on all three datasets. These results indicate that the target categories have little effect on the performance of ATSTA and the attacker can choose any target category to launch an attack.

C. Stealthiness of Backdoor Attacks

1) Human Studies: For the stealthiness of the attacks, we focus on the imperceptibility of the triggers. The research in the domain of image mostly focuses on the detection of the similarity between poisoned samples and benign samples [57], while they may easily misjudge audio triggers. For example, poisoned samples with bird chirps as triggers are detected by similarity recognition but do not appear suspicious to humans. Therefore, we test the stealthiness of the attacks through volunteers, and the experimental results are shown in Table IV.

The results of the experiment showed that 81.33% of the poisoned samples could be detected by the volunteers when using NBA. This is because although the whistling trigger used by NBA is a common environmental sound, it can still be perceived by volunteers during repeated tests. In contrast, only 18.67% and 9% of the poisoned samples were detected when using FlowMur and ATSTA. The possible reason for this is that FlowMur uses the strategy of dynamically injecting triggers, and there are some very low-amplitude regions in the audio samples, which make it easy for the trigger to be perceived by humans

TABLE V
EFFECT OF LOSS FUNCTION ON THE VALIDITY AND STEALTHINESS OF ATSTA (%)

λ		SCD		FKD		AudioMNIST		ER
λ_1	λ_2	BA	ASR	BA	ASR	BA	ASR	
0	0	97.42	99.95	97.93	99.96	98.90	100	22.67
	0.0001	97.11	99.04	98.72	98.19	98.26	98.78	16.33
	0.001	96.96	97.67	98.54	73.37	98.52	90.69	11
0.001	0	97.39	99.79	98.57	99.04	98.45	98.49	15.67
	0.0001	97.01	99.27	98.19	98.84	98.12	99.86	9
	0.001	97.34	98.45	98.54	84.52	98.49	94.11	5.67
0.01	0	97.48	97	98.19	92.87	99.61	96.95	7.67
	0.0001	96.24	95.79	96.92	93.61	99.42	95.93	1.33
	0.001	97.4	92.47	97.80	56.81	98.19	83.90	0

once it is added to the bass region of the sample. In contrast, ATSTA's maximum-peak trigger injection strategy ensures that the trigger is always hidden inside the maximum peak of the audio sample, so it is less likely to be detected by humans. In addition, it should be noted that our approach requires a lower injection rate compared to FlowMur, and this means its stealthiness is greater during the actual attack.

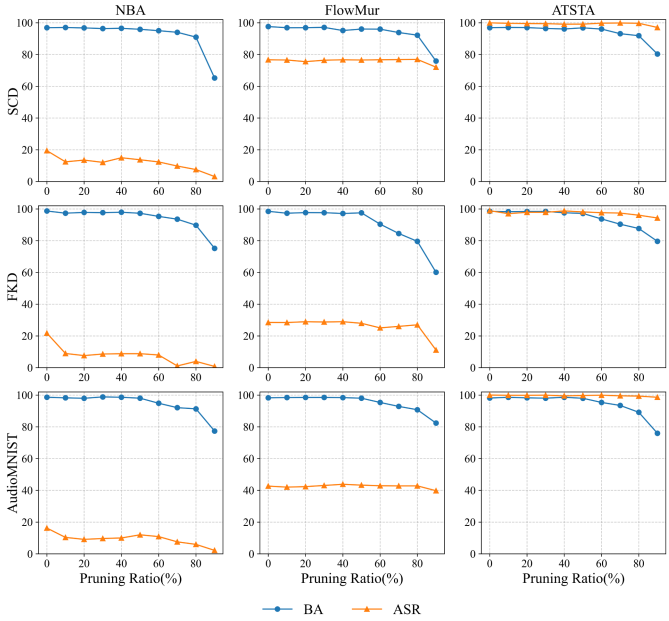

Fig. 6. Robustness of NBA, FlowMur and ATSTA to Fine-Pruning on the three datasets.

D. Ablation Experiments With Loss Functions

In this section, we study the effects of MSE loss and TV loss on ATSTA stealthiness and attack effectiveness in Section IV-C2. We control the MSE loss and TV loss through λ_1 and λ_2, respectively. Unless otherwise stated, all settings are consistent with those described in Section V-A5, and the experimental results are shown in Table V.

From this table, it can be seen that increasing the weight of both MSE loss and TV loss leads to a decrease in ASR and ER. Among them, TV loss has a greater impact on attack effectiveness and stealthiness, because TV loss smoothes the trigger in a way that enhances the imperceptibility to humans while causing some of the features that affect effectiveness to disappear. Meanwhile, MSE loss enhances the stealthiness of the trigger by reducing the difference between poisoned samples and benign samples, but it also leads to a decrease in ASR. Therefore, we need to determine the optimal loss weight combination to balance attack effectiveness and stealthiness. In particular, when $\lambda_1 = 0.001$ and $\lambda_2 = 0.0001$, ATSTA achieves more than 98% ASR on all three datasets and keeps the ER below 10%.

E. The Resistance to Defenses

We evaluate the robustness of ATSTA with the baseline method against three defense methods, Fine-pruning [54], STRIP [55] and Beatrix [56]. Unless otherwise stated, all other settings are the same as those shown in Section V-A5.

1) The Resistance to Fine-Pruning: As a classical backdoor defense method, Fine-Pruning aims at backdoor defense by pruning neurons that are rarely activated by benign samples. The rationale is that poisoned samples usually rely on triggers to activate neurons, resulting in different neurons activated by poisoned samples than by benign samples in the infected model. Specifically, the defender first sorts the neurons in the model in ascending order of their activation level in the face of benign samples, and then prunes out these neurons in that order.

The defense effect of Fine-Pruning is shown in Fig. 6. When using the NBA method, a decrease in ASR can be observed when the pruning rate is 10%, and BA starts to decrease when the pruning rate reaches 60%. This indicates that neurons activated by poisoned samples can be pruned before neurons activated by benign samples, and thus Fine-Pruning has a better defense against NBA. In contrast, the ASR on FlowMur and ATSTA only decreased slightly when the pruning rate reached over 80%, while BA gradually decreased from 50% pruning rate. This suggests that Fine-Pruning cannot distinguish between benign and infected neurons in FlowMur and ATSTA, namely FlowMur and ATSTA can bypass Fine-Pruning.

2) The Resistance to STRIP: The design concept of STRIP is that the trigger establishes a strong link between the poisoned samples and the target label. After adding a perturbation to a poisoned sample, the backdoor model will still predict the sample as the adversary's target class with high confidence. Based on this idea, STRIP superimposes other samples different from it on each input for perturbation and feeds these input samples with perturbations into the model to observe the prediction results. In this case, the predictive entropy of the poisoned samples is expected to be lower than the predictive entropy of the benign samples. Therefore, the defender can set an entropy threshold to identify whether the input samples carry triggers. The defense effect of STRIP is shown in Fig. 7.

As shown in Fig. 7, the predictive entropy distribution of poisoned samples and benign samples on NBA has a clear demarcation. The defender can distinguish between poisoned samples and benign samples by finding an appropriate threshold. Therefore, STRIP is effective for defense on NBA. However, the predictive entropy distributions of poisoned samples and

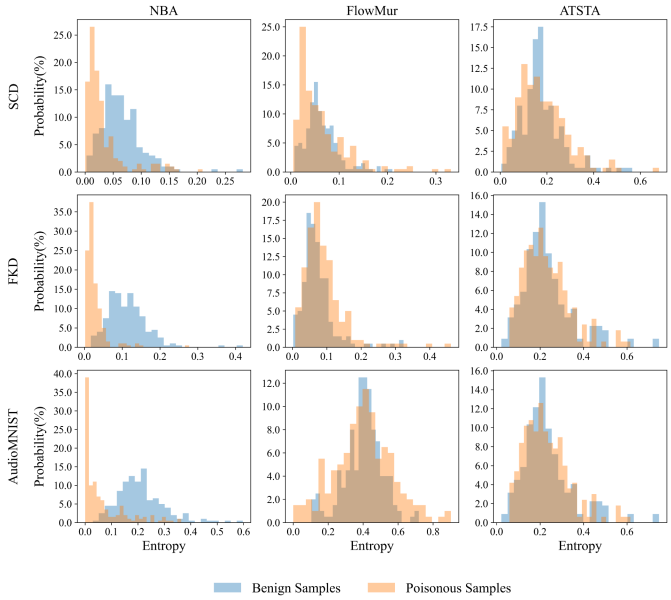

Fig. 7. Robustness of NBA, FlowMur and ATSTA to STRIP on the three datasets.

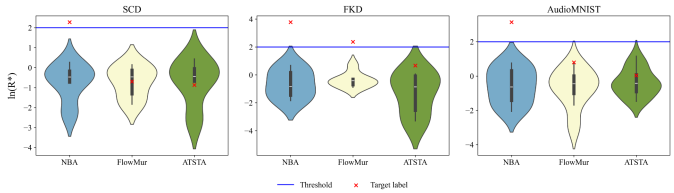

Fig. 8. Robustness of NBA, FlowMur and ATSTA to Beatrix on the three datasets.

benign samples on FlowMur and ATSTA have a large overlap. This suggests that STRIP fails to effectively distinguish between benign samples and poisoned samples under these two attacks. The possible reason is that the triggers generated by ATSTA become susceptible after the concealment constraint and are more vulnerable to perturbation. This proves that ATSTA has the ability to effectively resist STRIP.

3) The Resistance to Beatrix: The rationale behind Beatrix is that poisoned samples and benign samples do not intersect in the pixel space, and thus the intermediate representations of these can differ significantly. Based on this, Beatrix amplifies the difference between poisoned samples and benign samples through the Gram matrix, and uses a kernel-based test to identify target labels without making any a priori assumptions about the representation distribution. Fig. 8 illustrates the effectiveness of Beatrix against ATSTA and baseline on the three datasets.

As shown in Fig. 8, R^* is the anomaly index, which is consistent with the setting of Ma et al. [56], and we set its threshold to e^2. The R^* of the target category is above the threshold while the anomaly index of the non-target category does not exceed the threshold when using NBA. This shows that Beatrix can effectively defend against NBA. For FlowMur, Beatrix can successfully detect the target category on FKD, but performs poorly on SCD and AudioMNIST. In contrast, the target category of ATSTA cannot be detected by Beatrix on all the three datasets, proving the effective robustness of our method against Beatrix defense.

VI. Discussion

A. Possible Responses

In Section V-E, we analyze the resistance of ATSTA against common backdoor defenses. Among them, Fine-Pruning and STRIP rely on specific observations that do not apply to backdoor attacks using optimization-based triggers. Therefore, ATSTA can bypass them. In addition, Beatrix relies on the significance difference between poisoned samples and benign samples, but narrowing the difference between poisoned and benign samples by crafting constraints can cause this defense to fail. Furthermore, Beatrix requires the defender to have sufficient knowledge to acquire a portion of the benign samples, which limits its practical application. Based on the above analysis, we need to explore a possible defense strategy against ATSTA.

In Section V-D, we find that the ASR of poisoned samples after stealthiness constraints will decrease as the strength of the constraints increases, without significantly affecting BA. Therefore, we believe that mutual constraint between samples of the same class is a feasible defense scheme.

B. Limitations and Future Work

The performance of ATSTA on SCD, FKD, and AudioMNIST demonstrates its effectiveness in speech command recognition systems and speaker recognition systems. However, the applicability of ATSTA may be limited when facing more complex automatic audio recognition systems. This is because the higher complexity of automatic audio recognition systems challenges backdoor implantation. We propose a possible backdoor attack method against automatic audio recognition systems by modifying the generation model or constraints of triggers, which we regard as future work.

VII. Conclusion

In this paper, we propose an efficient and stealthy audio backdoor attack method, ATSTA, which only requires the attacker to possess a small set of target-class samples. ATSTA solves the current problem of limited-knowledge audio backdoor attacks that are highly dependent on the poisoning rate and have low attack efficiency. Specifically, we supplement the attacker's knowledge by constructing a surrogate dataset, then alternately training the model and optimizing triggers on that dataset, thus maximizing the attack capability of the trigger. We also design a trigger injection strategy based on maximum peaks and loss constraints, which enhances the stealthiness of ATSTA by masking the triggers through the peaks of speech signals and restricting the poisoned samples using two loss functions. Subsequently, on both speech command recognition systems and speaker recognition systems, we conduct a series of experiments on three datasets to validate the performance of ATSTA across three critical dimensions: attack effectiveness, stealthiness, and robustness against defenses.

The contents of this paper not only reveal the potential hazards of backdoor attacks in audio recognition systems, but also emphasize the urgency of implementing secure and reliable audio recognition systems in the real world. Through the analysis of this paper, we find that the proposed method is not only theoretically innovative, but also fully verified for its effectiveness and stealthiness in practical applications.

REFERENCES

[1] T. Ammari, J. Kaye, J. Y. Tsai, and F. Bentley, "Music, search, and IoT: How people (really) use voice assistants," *ACM Trans. Comput.- Hum. Interact.*, vol. 26, no. 3, pp. 1–28, 2019.

[2] M. Goldblum et al., "Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 1563–1580, Feb. 2023.

[3] Y. Li, Y. Jiang, Z. Li, and S.-T. Xia, "Backdoor learning: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 1, pp. 5–22, Jan. 2024.

[4] Y. Yao, H. Li, H. Zheng, and B. Y. Zhao, "Latent backdoor attacks on deep neural networks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2019, pp. 2041–2055.

[5] T. Gu, K. Liu, B. Dolan-Gavitt, and S. Garg, "BadNets: Evaluating backdooring attacks on deep neural networks," *IEEE Access*, vol. 7, pp. 47230–47244, 2019.

[6] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Statist.*, 2017, pp. 1273–1282.

[7] C. Wang et al., "Improving self-supervised learning for speech recognition with intermediate layer supervision," in *Proc. 2022 IEEE Int. Conf. Acoust., Speech Signal Process.*, 2022, pp. 7092–7096.

[8] B. Zhang et al., "WENETSPEECH: A 10000+ hours multi-domain mandarin corpus for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2022, pp. 6182–6186.

[9] B. Yan, R. Zhang, and Z. Yan, "VoiceSketch: A privacy-preserving voiceprint authentication system," in *Proc. IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun.*, 2022, pp. 623–630.

[10] R. Zhang, Z. Yan, X. Wang, and R. H. Deng, "VOLERE: Leakage resilient user authentication based on personal voice challenges," *IEEE Trans. Dependable Secure Comput.*, vol. 20, no. 2, pp. 1002–1016, Mar./Apr. 2023.

[11] R. Zhang, Z. Yan, X. Wang, and R. H. Deng, "LiVoAuth: Liveness detection in voiceprint authentication with random challenges and detection modes," *IEEE Trans. Ind. Informat.*, vol. 19, no. 6, pp. 7676–7688, Jun. 2023.

[12] Y. Zhang, M. Alder, and R. Togneri, "Using Gaussian mixture modeling in speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 1994, vol. 1, pp. I/613–I/616.

[13] K.-F. Lee, H.-W. Hon, M.-Y. Hwang, and X. Huang, "Speech recognition using hidden Markov models: A CMU perspective," *Speech Commun.*, vol. 9, no. 5/6, pp. 497–508, 1990.

[14] M. Gales et al., "The application of hidden Markov models in speech recognition," *Found. Trends Signal Process.*, vol. 1, no. 3, pp. 195–304, 2008.

[15] G. Hinton et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[16] J. S. Garofolo et al., "TIMIT acoustic-phonetic continuous speech corpus," Linguistic Data Consortium, 1993. [Online]. Available: https://catalog.ldc.upenn.edu/LDC93S1

[17] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2013, pp. 6645–6649.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[19] R. Vygon and N. Mikhaylovskiy, "Learning efficient representations for keyword spotting with triplet loss," in *Proc. 23rd Int. Conf. Speech Comput.*, St Petersburg, Russia: Springer, 2021, pp. 773–785.

[20] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digit. Signal Process.*, vol. 10, no. 1–3, pp. 19–41, 2000.

[21] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 5329–5333.

[22] X. Chang, W. Zhang, Y. Qian, J. Le Roux, and S. Watanabe, "End-to-end multi-speaker speech recognition with transformer," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 6134–6138.

[23] Y. Liu et al., "Trojaning attack on neural networks," in *Proc. 25th Annu. Netw. Distrib. Syst. Secur. Symp.*, Internet Soc, 2018, pp. 1–15.

[24] T. Zhai, Y. Li, Z. Zhang, B. Wu, Y. Jiang, and S.-T. Xia, "BackDoor attack against speaker verification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 2560–2564.

[25] S. Koffas, J. Xu, M. Conti, and S. Picek, "Can you hear it? Backdoor attacks via ultrasonic triggers," in *Proc. ACM Workshop Wireless Secur. Mach. Learn.*, 2022, pp. 57–62.

[26] Y. Luo, J. Tai, X. Jia, and S. Zhang, "Practical backdoor attack against speaker recognition system," in *Proc. Int. Conf. Inf. Secur. Pract. Experience*, 2022, pp. 468–484.

[27] J. Xin, X. Lyu, and J. Ma, "Natural backdoor attacks on speech recognition models," in *Proc. Int. Conf. Mach. Learn. Cyber Secur.*, 2022, pp. 597–610.

[28] J. Lan, J. Wang, B. Yan, Z. Yan, and E. Bertino, "FlowMur: A stealthy and practical audio backdoor attack with limited knowledge," in *Proc. IEEE Symp. Secur. Privacy*, 2024, pp. 1646–1664.

[29] Y. Kong and J. Zhang, "Adversarial audio: A new information hiding method and backdoor for DNN-based speech recognition models," 2019, *arXiv:1904.03829*.

[30] J. Ye, X. Liu, Z. You, G. Li, and B. Liu, "DriNet: Dynamic backdoor attack against automatic speech recognization models," *Appl. Sci.*, vol. 12, no. 12, 2022, Art. no. 5786.

[31] H. Cai, P. Zhang, H. Dong, Y. Xiao, S. Koffas, and Y. Li, "Towards stealthy backdoor attacks against speech recognition via elements of sound," *IEEE Trans. Inf. Forensics Secur.*, vol. 19, pp. 5852–5866, 2024.

[32] Y. A. Li, A. Zare, and N. Mesgarani, "StarGANv2-VC: A diverse, unsupervised, non-parallel framework for natural-sounding voice conversion," in *Proc. Interspeech*, 2021, pp. 1349–1353.

[33] Y. Zeng, M. Pan, H. A. Just, L. Lyu, M. Qiu, and R. Jia, "Narcissus: A practical clean-label backdoor attack with limited information," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2023, pp. 771–785.

[34] Y. Li, Y. Li, B. Wu, L. Li, R. He, and S. Lyu, "Invisible backdoor attack with sample-specific triggers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 16443–16452.

[35] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[36] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*. Palmdale, PA, USA: IGI Global, 2010, pp. 242–264.

[37] Y. Liu, L. Jin, and S. Lai, "Automatic labeling of large amounts of handwritten characters with gate-guided dynamic deep learning," *Pattern Recognit. Lett.*, vol. 119, pp. 94–102, 2019.

[38] B. Ma, Z. Tao, R. Ma, C. Wang, J. Li, and X. Li, "A high-performance robust reversible data hiding algorithm based on polar harmonic fourier moments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 4, pp. 2763–2774, Apr. 2024.

[39] C. Micheyl, K. Delhommeau, X. Perrot, and A. J. Oxenham, "Influence of musical and psychoacoustical training on pitch discrimination," *Hear. Res.*, vol. 219, no. 1/2, pp. 36–47, 2006.

[40] E. Zwicker and H. Fastl, *Psychoacoustics: Facts and Models*, vol. 22. Berlin, Germany: Springer 2013.

[41] G. Chen et al., "Who is real Bob? Adversarial attacks on speaker recognition systems," in *Proc. IEEE Symp. Secur. Privacy.*, 2021, pp. 694–711.

[42] Q. Li et al., "Concealed attack for robust watermarking based on generative model and perceptual loss," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 8, pp. 5695–5706, Aug. 2022.

[43] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[44] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2414–2423.

[45] C. Shi et al., "Audio-domain position-independent backdoor attack via unnoticeable triggers," in *Proc. 28th Annu. Int. Conf. Mobile Comput. Netw.*, 2022, pp. 583–595.

[46] J. Shore and R. Johnson, "Properties of cross-entropy minimization," *IEEE Trans. Inf. Theory*, vol. TIT-27, no. 4, pp. 472–482, Jul. 1981.

[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–15.

[48] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," 2018, *arXiv:1804.03209*.

[49] A. M. Rostami, A. Karimi, and M. A. Akhaee, "Keyword spotting in continuous speech using convolutional neural network," *Speech Commun.*, vol. 142, pp. 15–21, 2022.
[50] S. Becker, J. Vielhaben, M. Ackermann, K.-R. Müller, S. Lapuschkin, and W. Samek, "AudioMNIST: Exploring explainable artificial intelligence for audio analysis on a simple benchmark," *J. Franklin Inst.*, vol. 361, no. 1, pp. 418–428, 2024.
[51] G. E. Hinton et al., "Learning distributed representations of concepts," in *Proc. 8th Annu. Conf. Cogn. Sci. Soc.*, Amherst, MA, USA, 1986, vol. 1, pp. 1–12.
[52] S. Hochreiter, "Long short-term memory," in *Neural Computation*. Cambridge, MA, USA: MIT Press, 1997.
[53] A. Berg, M. O'Connor, and M. T. Cruz, "Keyword transformer: A self-attention model for keyword spotting," in *Proc. Interspeech*, 2021, pp. 4249–4253.
[54] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *Proc. Int. Symp. Res. Attacks, Intrusions, Defenses.*, 2018, pp. 273–294.
[55] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "Strip: A defence against trojan attacks on deep neural networks," in *Proc. 35th Annu. Comput. Secur. Appl. Conf.*, 2019, pp. 113–125.
[56] W. Ma, D. Wang, R. Sun, M. Xue, S. Wen, and Y. Xiang, "The 'Beatrix' resurrections: Robust backdoor detection via gram matrices," in *Proc. NDSS*, Internet Soc., 2023, pp. 1–18.
[57] X. Gong, Y. Chen, J. Dong, and Q. Wang, "ATTEQ-NN: Attention-based QoE-aware evasive backdoor attacks," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, Internet Soc., 2022, pp. 1–18.

Bo Wang (Member, IEEE) received the B.S. degree in electronic and information engineering, and the Ph.D. degree in signal and information processing from the Dalian University of Technology, Dalian, China, in 2003 and 2010, respectively. From 2010 to 2012, he was a Postdoctoral Research Associate with the Faculty of Management and Economics, Dalian University of Technology. From 2017 to 2019, he was with the State of University of New York at Buffalo, as a Visiting Scholar. He is currently a Professor with the School of Information and Communication Engineering, Dalian University of Technology. His research interests include artificial intelligence security and multimedia security.

Yiming Yan received the B.S. degree in electronic and information engineering from the Dalian University of Technology, Dalian, China, in 2023, where he is currently working toward the master's degree with the School of Information and Communication Engineering. His research focuses on image and audio oriented backdoor attacks and defenses.

Maozhen Zhang received the M.S. degree from the School of Computer Science and Technology, Dalian Maritime University, Dalian, China, in 2022. He is currently working toward the Ph.D. degree with the School of Information and Communication Engineering, Dalian University of Technology, Dalian. His research focuses on artificial intelligence security, with particular interests in federated learning, privacy security, deep learning, backdoor attacks and defenses.

Wei Wang (Member, IEEE) received the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China, in 2012. He is currently an Associate Professor with the New Laboratory of Pattern Recognition and the State Key Laboratory of Multimodal Artificial Intelligence Systems, CASIA. His research interests include artificial intelligence safety and multimedia forensics.